

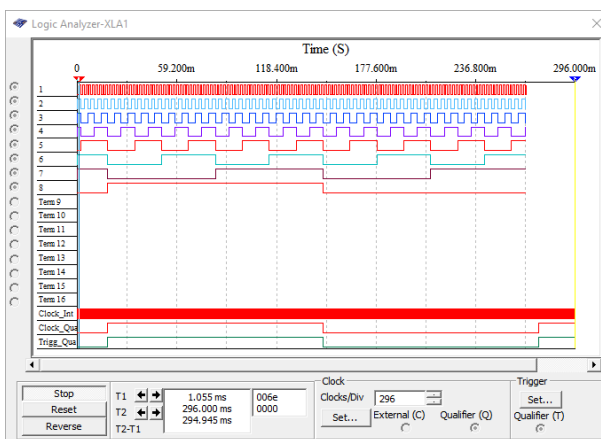
Міністерство освіти і науки України
Національний університет водного господарства та
природокористування

**Є.З. Маланчук, В.В. Макаренко,
В.М. Співак, Г.Г. Власюк,
А.В. Рудик**

МОДЕЛЮВАННЯ ТА АНАЛІЗ ЦИФРОВИХ СХЕМ

Підручник

*Затверджено вченою радою Національного університету
водного господарства та природокористування
як підручник для студентів вищих навчальних закладів,
які навчаються за спеціальностями 151 „Автоматизація та
комп'ютерно-інтегровані технології”, 141 „Електроенергетика,
електротехніка та електромеханіка” та
171 „Електроніка”*



УДК 621.382.2 / 3. (075.8)
ББК 32.844 1я73

Р83

*Затверджено вченою радою Національного університету
водного господарства та природокористування.
(Протокол № від 03.06.2017 р.)*

Рецензенти:

Мещанинов С.К., д-р. техн. наук, професор Дніпродзержинського державного технічного університету;

Гумен М.Б., канд. техн. наук, професор Національного авіаційного університету (м. Київ);

Древецький В.В., д-р. техн. наук, професор Національного університету водного господарства та природокористування (м. Рівне).

Є.З. Маланчук. Моделювання та аналіз цифрових схем. Підручник /
Є.З. Маланчук, В.В. Макаренко, В.М. Співак, Г. Г. Власюк, А.В. Рудик. – Рівне: НУВГП, 2018. – 463 с.

У підручнику розглянуто широкий круг питань, пов'язаних з вивченням, проектуванням, аналізом, моделюванням і використанням елементів та вузлів цифрових та імпульсних пристроїв за допомогою програми NI Multisim.

Посібник призначений для студентів технічних спеціальностей вищих навчальних закладів.

УДК 621.382.2 / 3. (075.8)
ББК 32.844 1я73

© Маланчук Є.З., Макаренко В.В, Співак В.М.,
Рудик А.В., Власюк Г. Г – Рівне: НУВГП,
2018. – 463 с.

© Національний університет водного
господарства та природокористування
© Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря
Сікорського»

ЗМІСТ

Вступ.....	7
1. Програма моделювання NI Multisim.....	9
1.1. Призначення програми NI Multisim.....	9
1.2. Початок роботи з програмою NI Multisim.....	12
1.3. Компоненти та прилади NI Multisim.....	14
Контрольні питання.....	28
2. Сигнали та методи їх дослідження.....	24
2.1. Загальні характеристики сигналів.....	24
2.2. Аналогові сигнали.....	27
2.3. Імпульсні сигнали.....	35
2.4. Вимірювання параметрів сигналів за допомогою програми NI Multisim.....	42
2.4.1. Джерела сигналів у Multisim.....	42
2.4.2. Інструменти аналізу сигналів у Multisim.....	44
2.4.3. Настроювання параметрів аналізу.....	49
2.5. Основні правила роботи з вимірювальними приладами.....	55
2.5.1. Осцилограф.....	55
2.5.2. Плотер Боде.....	58
2.5.3. Генератор слова.....	64
2.5.4. Аналізатор спектру.....	69
2.5.5. Логічний аналізатор.....	72
Контрольні питання.....	79
3. Проходження імпульсних сигналів через RC-кола.....	80
3.1. Диференціальні, інтегровальні та розділові RC-кола.....	80
3.2. Розрахунок лінійного формувача імпульсів.....	92
3.3. Обмежувачі послідовного і паралельного типу на діодах.....	97
Контрольні питання.....	103
4. Транзисторні насичені ключі.....	104

4.1. Призначення транзисторних ключів.....	104
4.2. Лінійні моделі транзисторів в режимі великого сигналу.....	104
4.3. Методика розрахунку параметрів транзисторного ключа.....	109
4.4. Аналіз перехідних процесів в транзисторному ключі....	114
Контрольні питання.....	127
5. Арифметичні основи цифрової техніки.....	128
5.1. Перетворення чисел між системами числення.....	128
5.2. Інформаційна ємність та форми зображення двійкових чисел.....	135
5.3. Цифрові коди.....	142
5.4. Двійкова арифметика.....	153
Запитання та вправи.....	164
6. Логічні основи цифрової техніки.....	167
6.1. Алгебра логіки.....	167
6.2. Мінімізація логічних функцій.....	188
6.3. Основи схемної реалізації логічних функцій.....	198
6.4. Проектування логічних схем.....	216
6.5. Приклад логічного проектування.....	228
Запитання та вправи.....	236
7. Базові елементи цифрових інтегрованих мікросхем.....	240
7.1. Базові елементи ТТЛ.....	240
7.2. Базові елементи ЕСЛ.....	267
7.3. Базові елементи на МОН-структурах.....	276
7.4. Базові елементи І ² Л.....	287
Запитання та вправи.....	291
8. Комбінаційні схеми.....	293
8.1. Дешифратори.....	293
8.2. Шифратори.....	298
8.3. Мультиплексори.....	299
8.4. Демультиплексори.....	301
8.5. Схемна реалізація логічних функцій на дешифраторах мультиплексорах.....	303

8.6. Суматори кодів.....	307
8.7. Схеми порівняння кодів або цифрові компаратори.....	311
8.8. “Небезпечні змагання” сигналів в комбінаційних пристроях.....	313
Контрольні питання.....	315
9. Тригерні елементи.....	317
9.1. Призначення та класифікація тригерів.....	317
9.2. Асинхронні та синхронні RS-тригери.....	319
9.3. Тригери Т-типу.....	323
9.4. Тригери D-типу.....	324
9.5. Тригери JK-типу.....	326
9.6. Універсальні RSD- та RSJK-тригери.....	327
9.7. Приклади використання тригерів.....	328
Контрольні питання.....	335
10. Регістри.....	336
10.1. Загальні зауваження.....	336
10.2. Паралельні регістри на D-тригерах.....	336
10.3. Послідовні регістри.....	338
10.4. Паралельно-послідовні та послідовно-паралельні регістри.....	341
10.5. Послідовні регістри зі зворотними зв'язками.....	342
10.6. Приклади використання регістрів.....	344
10.7. Проектування цифрового послідовнісного пристрою на регістрі зсуву.....	349
Контрольні питання.....	351
11. Лічильники імпульсів.....	353
11.1. Призначення та класифікація лічильників імпульсів... 353	
11.2. Двійкові лічильники.....	354
11.3. Реверсивні лічильники.....	357
11.4. Синхронні двійкові лічильники.....	358
11.5. Недвійкові лічильники.....	359
11.6. Лічильники на регістрах зсуву.....	372
11.7. Застосування регістрів та лічильників для побудови схем затримки імпульсних сигналів.....	373

11.8. Проектування недвійкових лічильників на тригерах та мікросхемах двійкового лічильника.....	375
Контрольні питання.....	385
12. Генератори імпульсів.....	387
12.1. Загальна характеристика генераторів імпульсів.....	387
12.2. Очікувальні мультівібратори (одновібратори).....	389
12.3. Мультівібратори.....	396
13. Запам'ятовувальні пристрої.....	408
13.1. Оперативні запам'ятовувальні пристрої.....	408
13.2. Мікросхеми масочних ПЗП.....	414
13.3. Мікросхеми програмованих ПЗП.....	417
13.4. Мікросхеми репрограмованих ПЗП.....	422
Контрольні питання.....	428
14. Застосування цифрових інтегральних мікросхем.....	429
14.1. Завади та завадостійкість цифрових пристроїв.....	429
14.2. Рекомендації з монтажу інтегральних мікросхем.....	438
Контрольні питання.....	451
Перелік літератури.....	453

ВСТУП

Значний прогрес у розвитку багатьох областей науки й техніки обумовлений розвитком електроніки. Сьогодні неможливо знайти галузь промисловості, у якій не використовувалися б електронні прилади або електронне обладнання вимірювальної техніки, автоматики й обчислювальної техніки. При цьому тенденція розвитку така, що частка електронних інформаційних та автоматичних пристроїв безупинно збільшується. Це є результатом розвитку інтегральної технології, впровадження якої дозволило налагодити масовий випуск дешевих та високоякісних мікроелектронних функціональних вузлів різного призначення.

На сьогоднішній день промисловість випускає велику кількість електронних функціональних вузлів, необхідних для створення пристроїв автоматики, вимірювальної й обчислювальної техніки, перетворювачів інформації та ін.: інтегральні підсилювачі, комутатори, логічні елементи, лічильники, регістри, аналого-цифрові та цифро-аналогові перетворювачі, формувачі та ін. Типові електронні вузли дозволяють зібрати потрібний електронний блок без детального розрахунку його окремих складових частин.

Необхідно тільки вірно підібрати інтегральні мікросхеми, розробити схему їх з'єднання та ввести зворотні зв'язки необхідного виду. Але для цього потрібно розуміти не тільки принципи функціонування окремих елементів, а і принципи побудови складних пристроїв та систем, у яких взаємодіє велика кількість елементів. До того ж велика кількість пристроїв і систем містить як аналогові, так і цифрові компоненти, що потребує узгодження їх параметрів.

Дисципліна «Цифрова схемотехніка» є базовою дисципліною і забезпечує вивчення схемотехніки цифрових пристроїв та їх використання в різних модулях електронних систем. Дисципліна спрямована на вивчення фізичних основ перетворення аналогових сигналів в цифрову форму і обробки цифрових сигналів на основі методів і засобів дискретної математики, а також пристроїв, що реалізують ці задачі.

Для проектування навіть нескладних електронних при-

строїв необхідне розуміння процесів, що відбуваються як в окремих його частинах, так і в пристрої в цілому. Зважаючи на те, що більшість студентів не можуть перевірити роботу електронних пристроїв на фізичних макетах, для засвоєння принципів роботи основних цифрових, аналогових та аналого-цифрових пристроїв доцільно проводити моделювання їх роботи за допомогою програми-симулятора.

Найбільш зручною з погляду інтерфейсу користувача є програма схемотехнічного моделювання NI Multisim. Вона дозволяє студентам закріпити матеріал, який вивчається в рамках курсу, отримати навички роботи з вимірювальною апаратурою різного призначення (хоча й віртуальною) і засвоїти методикку організації й проведення експериментів відповідно до поставленого завдання.

Спеціально для компанії Analog Devices компанією National Instruments була розроблена безкоштовна версія програми, у якій дещо обмежена кількість компонентів у базі користувача й встановлено обмеження на число компонентів у кількості 25, які можуть бути використані для створення схеми.

Як показує практика, такої кількості елементів в схемі достатньо для того, щоб створити кожний з пристроїв, які вивчаються в базових курсах схемотехніки.

Цим і пояснюється структура підручника. Спочатку відбувається знайомство з основами роботи в програмі схемотехнічного моделювання, а потім розглядаються основні розділи дисципліни «Цифрова схемотехніка» з ілюстрацією основних положень за допомогою моделей, що можуть бути легко повторені навіть малокваліфікованим користувачем.

В результаті вивчення курсу «Цифрова схемотехніка» студент має знати: принципи побудови та функціонування цифрових та аналого-цифрових пристроїв; принципи вибору методів аналізу та розрахунку електронних пристроїв із заданими характеристиками, а також вміти: розраховувати електронні кола; узагальнювати вплив характеристик окремих вузлів електронних пристроїв на їх параметри; виконувати синтез різних електронних пристроїв та проводити їх моделювання.

1. ПРОГРАМА МОДЕЛЮВАННЯ NI MULTISIM

1.1. Призначення програми NI Multisim

Розробка будь-якого радіоелектронного пристрою супроводжується фізичним або математичним моделюванням. Фізичне моделювання пов'язане зі значними матеріальними витратами, оскільки потрібно виготовляти макет і проводити досить трудомісткі дослідження. Найчастіше фізичне моделювання неможливе, наприклад, при розробці великих інтегральних схем (BIC). Тому при проектуванні доцільно застосовувати програми математичного моделювання, які не тільки дозволяють знизити строки й вартість розробки, але й надають розроблювачу цілий ряд інструментів аналізу, які або недоступні при фізичному моделюванні, або їх вартість надзвичайно висока.

На сьогоднішній день розроблювачі використовують кілька програм схемотехнічного моделювання: Micro-Cap, Alpak, Orcad, Altium Designer (раніше Protel), Proteus, Multisim (раніше Electronics Workbench), Labview та інші [1].

Найбільш зручними для моделювання є два пакети програм – Orcad та NI Multisim (це не відноситься до НВЧ-пристроїв). Безкоштовна версія програми моделювання NI Multisim Analog Devices Edition версії 10.0.1 призначена для моделювання схем з використанням аналогових компонентів, що випускаються компанією Analog Devices. Слід відмітити, що ліцензійна версія програми NI Multisim без пакетів розширення коштує більше 4000\$, а повна версія близько 6000\$. На сьогоднішній день вже випущена версія програми 3.01.

З програмою NI Multisim™ Analog Devices® Edition, яка поширюється вільно, компанія Analog Devices і підрозділ Electronics Workbench Group компанії National Instruments надають інженеру можливість розробки й моделювання різних радіоелектронних пристроїв: аналогових, цифрових і комбінованих аналого-цифрових.

До повної версії пакету NI Multisim входять програми NI

Ultiboard (для розробки друкованих плат, виконує розміщення й трасування з'єднань на платах, що мають до 32 шарів) і NI Multisim MCU Module (для налагодження та моделювання мікропроцесорних систем). Крім того, знімається багато обмежень, які введені в безкоштовній версії (наприклад, на число елементів схеми, яке в безкоштовній версії не має перевищувати 25).

Програма NI Multisim™ Analog Devices® Edition є Spice симулятором і дозволяє:

- моделювати аналогові схеми, використовуючи більше 800 бібліотечних елементів – операційних підсилювачів, ключів, комутаторів та ін.;
- досліджувати схеми з числом елементів не більше 25;
- використовувати для аналізу вбудовані інструменти та алгоритми, включаючи аналіз найгіршого випадку (що передбачає аналіз самого несприятливого варіанту умов);
- проводити заміну компонентів і зміну їх параметрів для детального вивчення властивостей схеми;
- здійснювати online зв'язок з Analog Devices Design Center для отримання додаткових інструментів аналізу;
- негайно отримувати доступ до довідкових даних будь-яких компонентів Analog Devices.

Програму Multisim (раніше Electronics Workbench) розробила канадська компанія Electronics Workbench (колишня Interactive Image Technologies). Головний продукт компанії Electronics Workbench, Multisim – це один з кращих інструментів для розрахунку лінійних і нелінійних електричних кіл і розробки друкованих плат. Серед користувачів програми такі відомі корпорації як Sony, Boeing, а також передові академічні установи на зразок Масачусетського технологічного інституту. В 2005 р. компанія Electronics Workbench була придбана американською корпорацією National Instruments з метою розвитку більш тісного зв'язку між вимірювальним устаткуванням, що використовується при створенні електронних пристроїв, і програмами, призначеними для розрахунку електричних кіл. Угода відбулася після того, як кілька років National Instruments співробітничала з Electronics Workbench для інтеграції програми Multisim у графічне середовище «віртуальних приладів» NI Labview. Незважаю-

чи на укладений контракт, компанія Electronics Workbench залишається незалежним підрозділом у складі National Instruments і в найближчі кілька років буде незалежно поширювати програми через власну дистрибуторську мережу.

Основні переваги програми Multisim:

1. Простий графічний редактор, що дозволяє досить просто малювати на екрані електронні схеми у звичному зображенні.

2. Велика бібліотека сучасних електронних компонентів (дискретних, інтегральних аналогових, цифрових і змішаних аналого-цифрових). Бібліотека відкрита, легко може поповнюватися новими елементами, у тому числі й вітчизняними.

3. Велика бібліотека моделей електронних схем, що дозволяє використовувати готові практичні розробки й легко модернізувати їх під конкретне завдання. Приклади схем, пропоновані компаніями National Instruments і Analog Devices, можна знайти в [2].

4. Широкий вибір віртуальних вимірювальних приладів, що дозволяє виконати будь-яке вимірювання (при цьому не тільки електричне вимірювання). Робота з цими віртуальними вимірювальними приладами максимально наближена до роботи з реальними приладами. Підключивши віртуальний прилад до будь-якої точки схеми, можна отримати вичерпну інформацію про процеси в даному вузлі.

5. Простий по інтерфейсу набір моделюючих засобів, що дозволяє крім традиційного моделювання електронної схеми на постійному та змінному струму провести моделювання спектральних, нелінійних, амплітудно-частотних (АЧХ) та фазочастотних (ФЧХ) характеристик, оцінити вплив температури на окремі компоненти й на схему в цілому, можливість зміни будь-яких параметрів компонентів, а також параметрів джерел сигналів і живлення. Досить просто можна виконати імовірнісний аналіз роботи схеми з різними законами розподілу параметрів.

6. Можливість підключення віртуальних приладів програми Labview.

7. Широкі можливості документування дослідження, отримання твердої копії як електричної схеми, так і параметрів

моделювання, а також інформації з екрану вимірювальної апаратури (якісно оформлених графічних результатів дослідження).

8. Низькі вимоги до швидкодії комп'ютера (можлива робота починаючи з 386 моделі процесора).

9. Не вимагає знань програмування. Потрібно лише знайомство із середовищем Windows. Інтуїтивний інтерфейс дозволяє швидко навіть непідготовленому користувачеві (буквально за півгодини) ознайомитися з основами та приступити безпосередньо до досліджень (електронного моделювання).

10. Ретельно підготовлений довідковий матеріал (Help) забезпечує контекстну допомогу як по меню, компонентам і опціям моделювання, так і по загальних питаннях моделювання.

Multisim забезпечує введення електричних схем для їхнього подальшого моделювання та дозволяє створювати текстовий опис цифрових пристроїв на мовах VHDL і Verilog для синтезу ПЛІС.

1.2. Початок роботи з програмою NI Multisim

Роботу з програмою слід почати з установки основних параметрів. Для цього необхідно в меню Options вибрати пункт Global Preferences (рис. 1.1).

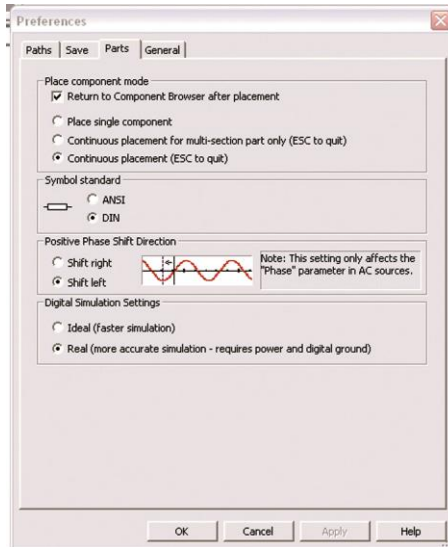


Рис. 1.1. Вікно установки основних параметрів програми

Відкривається вікно, у якому є чотири вкладки. У вкладці Parts вибираються:

- спосіб відображення елементів схеми – у стандарті DIN або ANSI;
- спосіб вводу елементів схеми;
- одиночний елемент;
- ввід всіх частин елементу (наприклад, ІС, до складу якої входять чотири операційні підсилювачі (ОП));
- ввід елементів необмежену кількість разів при натисканні лівої кнопки «миші»;
- аналіз схеми з використанням ідеальних або реальних моделей елементів.

У вкладці Paths задаються шляхи до файлів і робочої папки програми, а у вкладці Save задається автозбереження (autobackup), створення прихованої копії (для відновлення останньої працездатної версії досліджуваної схеми у випадку пошкодження файлу схеми) і режим збереження даних симулювання (моделювання) разом з параметрами вимірювальних приладів.

Додатково встановлюються параметри для робочого ли-

ста програми через меню Options/Sheet Properties (рис. 1.2). Тут можна задати колір фону та колір елементів. Доступними є 5 варіантів установки параметрів: білий або чорний фон та колір елементів і ліній за замовчуванням, чорний фон та білі елементи й лінії, білий фон та чорні елементи і лінії.

В режимі Custom користувач може задати колір усіх елементів та фону самостійно (рис. 1.3). У вкладці Sheet Properties задаються товщина ліній, режим відображення сітки, вибирається розмір і тип шрифту, розмір і орієнтація робочого листа й ще цілий ряд параметрів. Докладно з пунктами меню та їх призначенням можна ознайомитися за допомогою підказки програми (англ. Мовою).

Настроїти вигляд робочого вікна та панелі інструментів легко. Для цього досить помістити курсор «миші» в область листа з панелями інструментів і натиснути праву клавішу «миші». У контекстному меню, що відкриється, необхідно поставити галочки біля тих панелей інструментів, які бажано відображати.

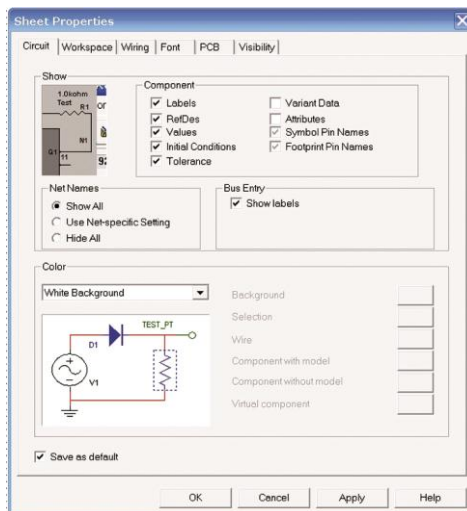


Рис. 1.2. Вікно установки параметрів робочого листа

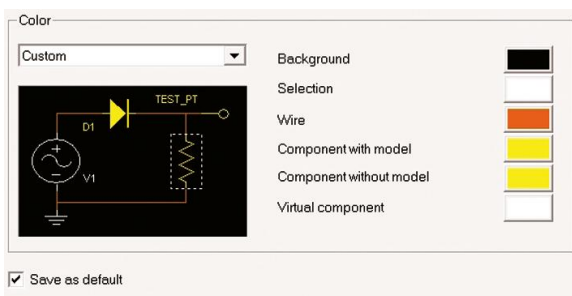


Рис. 1.3. Установка кольору елементів робочого листа

1.3. Компоненти та прилади NI Multisim

Після установки параметрів програми та робочого листа можна приступати до складання принципової схеми.

Для цього спочатку розміщують елементи схеми на робочому листі. Для розміщення елемента досить клацнути лівою кнопкою «миші» на піктограмі потрібної бібліотеки (рис. 1.4) або через меню Place/Component вибрати необхідний елемент. За замовчуванням панель компонентів відображається у вікні програми (рис. 1.5). Користувачу доступні бібліотеки джерел живлення, генераторів сигналів, пасивних компонентів, транзисторів і діодів, логічних і аналогових мікросхем, змішаних аналого-цифрових і цифро-аналогових компонентів, індикаторів, запобіжників, джерел опорної напруги, електромеханічних компонентів (двигунів, механічних контактів, котушок, трансформаторів та ін.).

Деякі елементи бібліотек джерел потужності та генераторів сигналів наведені в табл. 1.1.

На рис. 1.6 наведений приклад вікна програми при виборі резистора. У вікні Component вибирається номінальне значення опору резистора, у вікні Component type – тип виконання резистора: плівковий, провідний, композитний, для поверхневого монтажу (різного типу) та ін. Точність резистора (відхилення від номінального значення) вибирається у вікні Tolerance (%) з ряду 0, 0.1, 0.2, 0.5, 1 та 2.

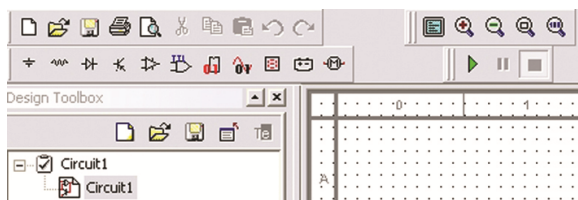


Рис. 1.4. Фрагмент робочого вікна програми з бібліотеками елементів

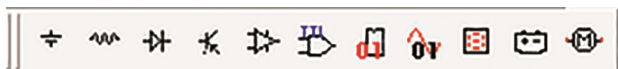











Рис. 1.5. Панель компонентів

При виборі інших компонентів (наприклад, змінного резистора або конденсатора) будуть з'являтися видозмінені вікна вибору параметрів компонентів.

Розглянемо на прикладі фільтра нижніх частот (ФНЧ) порядок роботи з програмою, вибір віртуальних приладів і аналіз отриманих результатів. Для аналізу виберемо схему, синтезовану за методикою, викладеною в [3], і розраховану за допомогою програми Filter Wiz Pro. Вихідні дані для проектування фільтра: нерівномірність АЧХ у смузі пропускання $a_c \leq 1$ дБ, гарантоване затухання в смузі затримування $a_s \geq 30$ дБ, частота зрізу фільтра $f_c = 3400$ Гц, частота гарантованого затухання 9000 Гц. При розрахунках отримано порядок фільтра $n = 3$ для апроксимації за Чебишевим та $n = 5$ для апроксимації за Батервортом. Вибираємо апроксимацію за Чебишевим.

Схема реалізується послідовним з'єднанням кола першого порядку з добротністю 0.5, під'єданого через повторювач напруги, і другого порядку, реалізованого за схемою Саллен-Кі, що реалізує добротність полюсу, яка дорівнює 2,02. Розрахована програмою Filter Wiz Pro АЧХ фільтра наведена на рис. 1.7, а значення опорів резисторів і ємностей конденсаторів наведені на електричній схемі (рис. 1.8). Колір провідників і компонентів на схемі залишені встановленими в програмі за замовчуванням. Слід зауважити, що набір резисторів, пропонує в меню програми, відповідає ряду E96, і на схемі розміщені компоненти відповідно до цього ряду, хоча розрахункові значення резисторів становлять: $R1 = 94,72$ кОм, $R2 = R3 = 99,89$ кОм.

Табл. 1.1. Елементи бібліотек джерел живлення та генераторів сигналів

POWER_SOURCES							SIGNAL_SOURCES	
DC_ Power	DGND	Ground	VCC	VDD	VEE	VSS	Clock_ Voltage	AC_ Voltage
								
Джерело постійної напруги	Цифровий спільний	Спільний	Джерело +Uж для ІМС TTL	Джерело +Uж для ІМС КМОП	Джерело -U-	Джерело живлення КМОП	Генератор імпульсів позитивної полярності	Генератор змінної напруги

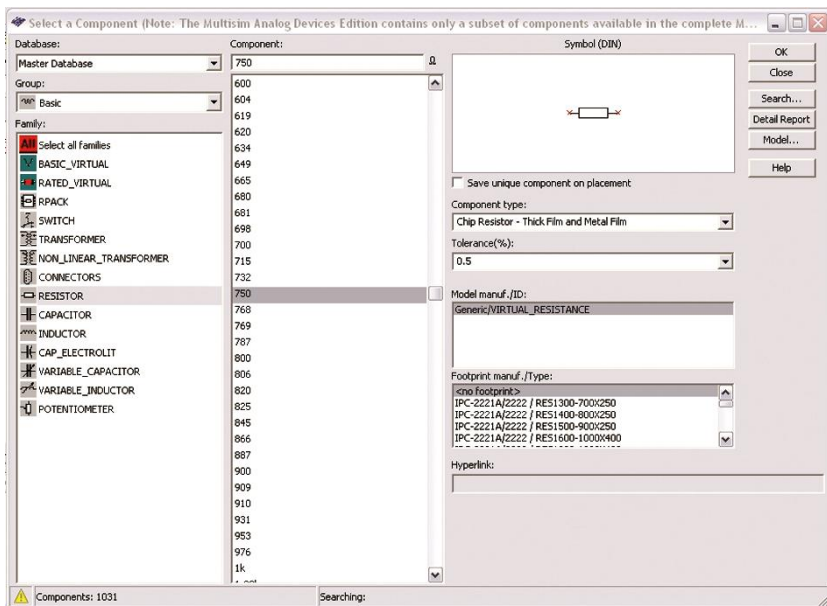


Рис. 1.6. Вікно вибору пасивних компонентів

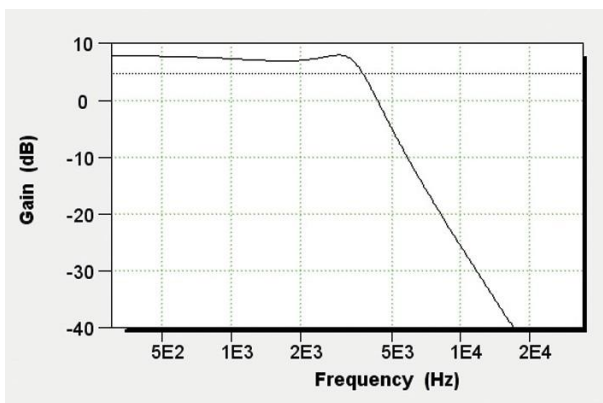


Рис. 1.7. Розрахована АЧХ фільтра нижніх частот

За необхідністю можна ввести резистори необхідного номіналу вручну. Для реалізації фільтра буде потрібно використати два операційні підсилювачі зі смугою одиничного підсилення (GBW – Guaranteed Bandwidth) не менш 1 МГц. Вибирає-

При розміщенні елементів на схемі зручно користуватися контекстним меню, яке викликається натисканням правої кнопки "м«ші»". Воно дозволяє повертати елементи за і проти годинникової стрілки на 90 градусів, здійснювати дзеркальні перетворення по вертикалі та горизонталі й містить ще багато різних пунктів. При підключенні джерел живлення (меню Place Source, яке позначено піктограмою спільного проводу (табл. 1.1) слід врахувати, що джерело позитивної напруги позначається як VCC, а негативної – як VEE.

Кожний з віртуальних приладів має багато параметрів, з якими знайомляться за необхідністю. Для аналізу схеми обмежимося поки трьома приладами:

- 19

- двоканальним осцилографом для аналізу форми сигналу на вході та виході фільтра;
- графопобудовником АЧХ і ФЧХ (Bode Plotter).

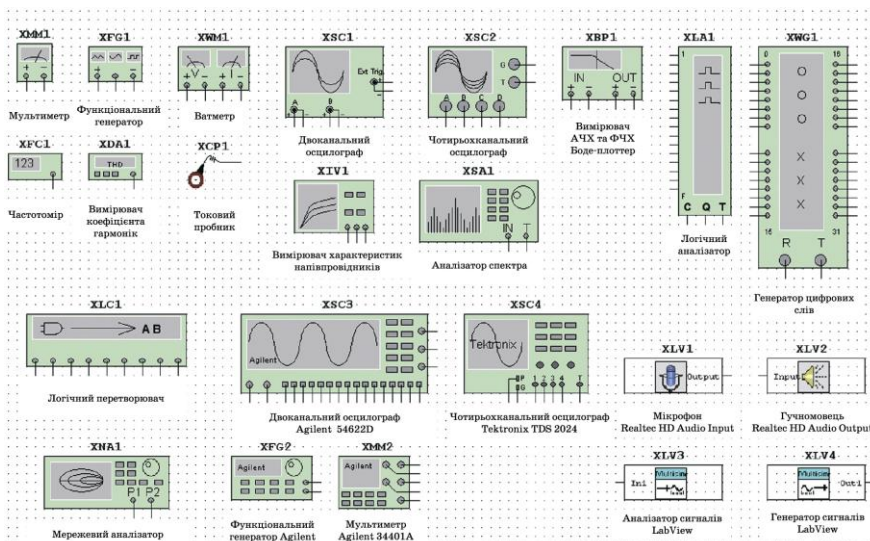


Рис. 1.9. Віртуальні вимірювальні прилади програми Multisim

Для розміщення приладу досить клацнути лівою кнопкою "м«ші" »а піктограмі приладу, розміщеної з правого боку робочого вікна програми, і перетягнути зображення в потрібне місце робочого листа.

Для того, щоб на екрані осцилографа вхідний і вихідний сигнали відображалися різним кольором, необхідно поміняти колір лінії, яка підключена до другого входу осцилографа. Для цього виділяють лінію клацанням лівої кнопки "м«ші" » потім натискають праву кнопку. У контекстному меню, що з'явився, потрібно вибрати пункт Change Color. Схема з підключеними приладами показана на рис. 1.10.

Після цього можна задати параметри тестового сигналу. Для цього необхідно клацнути на зображенні приладу лівою кнопкою "м«ші" »У вікні, що відкрилося, встановлюємо параметри сигналу на виході функціонального генератора (рис. 1.11). Можна вибрати одну з таких форм сигналу: синусоїдальна (гар-

монічна) напруга, послідовність трикутних імпульсів, послідовність прямокутних імпульсів. Частота сигналу задається в рядку Frequency в діапазоні від 1 ГГц до 103 ТГц, а амплітуда – в діапазоні від 1 фВ до 103 ТВ. Напруга зсуву задається у тому ж діапазоні. За замовчуванням на виході генератора формується двополярна напруга. Наприклад, якщо задана амплітуда 10 В для гармонічного сигналу, то на виході генератора буде сформовано сигнал з позитивною і негативною амплітудами 10 В. Якщо потрібно сформувати однополярний сигнал (наприклад, прямокутні імпульси амплітудою 5 В), то потрібно задати амплітуду сигналу на рівні половини від необхідної (для нашого прикладу 2,5 В) і задати позитивний зсув напруги +2,5 В.

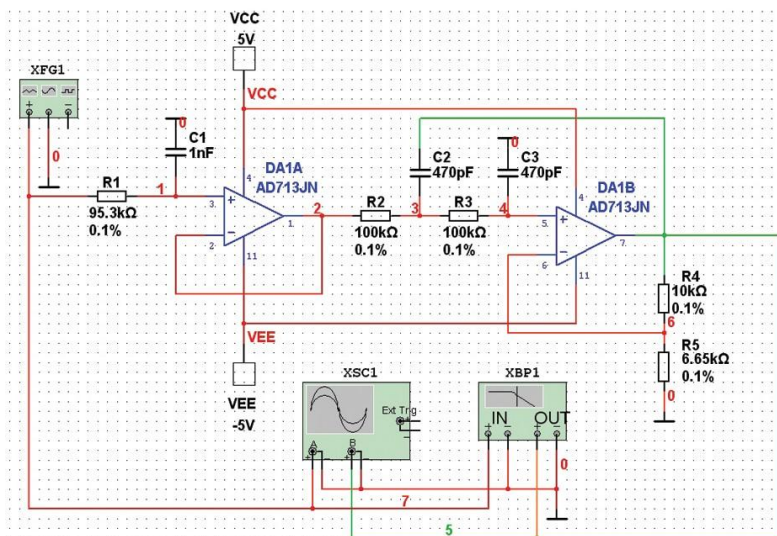


Рис. 1.10. Схема ФНЧ з підключеними вимірвальними приладами

Далі приступаємо до аналізу роботи схеми. Для цього достатньо або натиснути на зелений трикутник (рис. 1.4), або натиснути вимикач, розташований у правій верхній частині робочого вікна програми, або через меню Simulate викликати команду Run.

Проконтролюємо форму сигналу на вході і виході фільтра за допомогою осцилографа. Тому що коефіцієнт підсилення

фільтра (з розрахунку) дорівнює 2.5, а розмах вхідної напруги – 1 В, встановимо чутливість каналу А осцилографа 1 В/поділку, а каналу В – 2 В/поділку. Форма сигналу на вході та виході ФНЧ показана на рис. 1.12. Для рознесення сигналів на екрані осцилографа по вертикалі вводимо зсув Y position в каналі А на рівні +1.6 В, а в каналі В – -1.4 В.

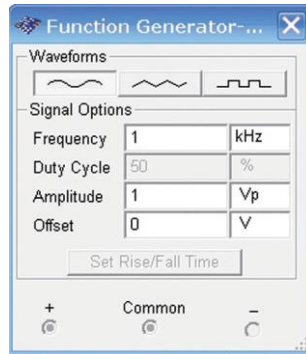


Рис. 1.11. Установка параметрів вхідного сигналу

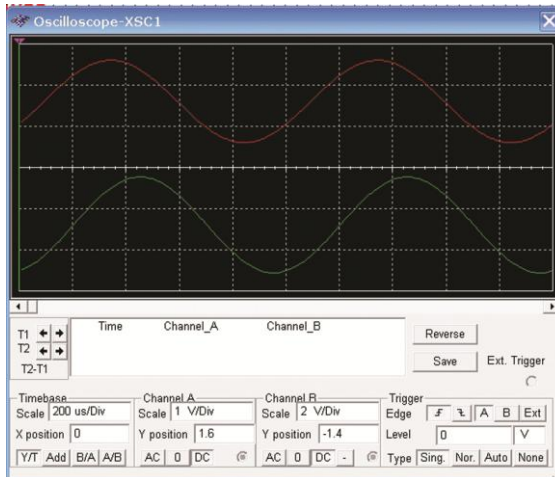


Рис. 1.12. Форма сигналу на вході (червоний) і виході (зелений) ФНЧ

Після цього можна здійснити перевірку АЧХ і ФЧХ спроектованого фільтра за допомогою Bode Plotter. Отримана в результаті моделювання АЧХ наведена на рис. 1.13, а ФЧХ – на

рис. 1.14. Крім того, можна зберегти файл з результатами визначення АЧХ і ФЧХ. Файл з розширенням .bod є текстовим файлом, фрагмент якого показаний на рис. 1.15. Аналіз результатів показав, що максимальна нерівномірність АЧХ в смузі пропускання небагато менше 1 дБ, що цілком задовольняє поставлені умови.

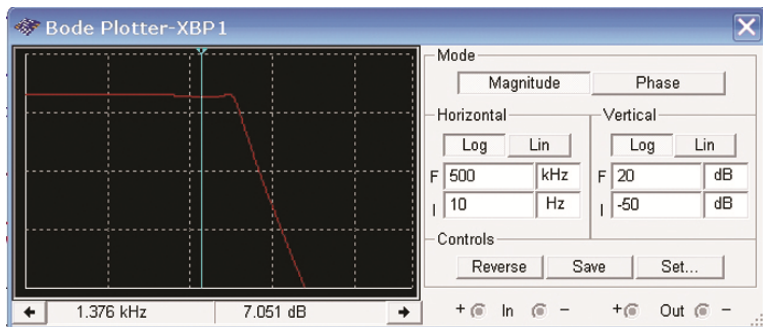


Рис. 1.13. Форма АЧХ змодельованого ФНЧ

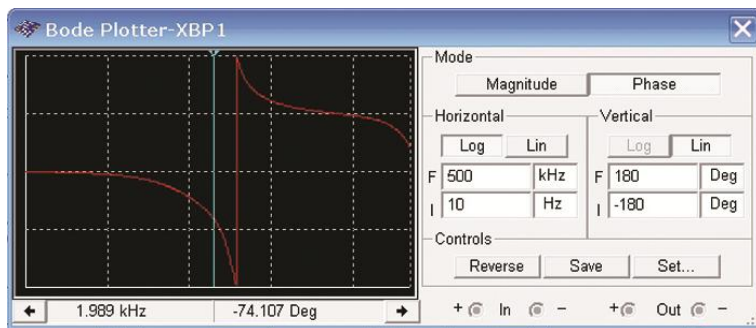


Рис. 1.14. Форма ФЧХ змодельованого ФНЧ

З безкоштовною версією програми поставляється один приклад (Getting Started), що складається з трьох різних файлів: лист з розміщеними компонентами (Getting Started 1), компоненти, з'єднані в схему (Getting Started 2), і схема з підключеними вимірювальними приладами (Getting Started 3). Використовуючи ці файли, легко провести експерименти із запропонованою в прикладі схемою, що дозволить швидше засвоїти роботу з програмою.

```

Bode data: BOD

column 1 Frequency (Hz)
column 2 Gain (dB)
column 3 Gain (Linear)
column 4 Phase (Deg)
trace name: Bode Result

Color: 255
Gain_Range_Start: 3.162278e-003
Gain_Range_End: 1.000000e+001
Phase_Range_Start: -1.800000e+002
Phase_Range_End: 1.800000e+002
Frequency Gain (dB) Gain Phase
-----
1.000000e+001 7.97149e+000 2.50365e+000 -4.27497e-001
1.02329e+001 7.97148e+000 2.50365e+000 -4.37454e-001
1.04713e+001 7.97148e+000 2.50365e+000 -4.47644e-001
1.07152e+001 7.97148e+000 2.50365e+000 -4.58071e-001
1.09648e+001 7.97147e+000 2.50365e+000 -4.68740e-001
1.12202e+001 7.97147e+000 2.50365e+000 -4.79658e-001
1.14815e+001 7.97146e+000 2.50365e+000 -4.90831e-001
1.17490e+001 7.97146e+000 2.50365e+000 -5.02263e-001
1.20226e+001 7.97145e+000 2.50364e+000 -5.13962e-001
1.23027e+001 7.97144e+000 2.50364e+000 -5.25934e-001
1.25893e+001 7.97144e+000 2.50364e+000 -5.38184e-001
-----

```

Рис.1.15. Фрагмент текстового файлу з результатами моделювання

Контрольні питання

1. Як настроїти інтерфейс програми NI Multisim?
2. Як в програми NI Multisim встановити розмір робочого листа та одиниць виміру?
3. Як в програми NI Multisim настроїти автоматичне збереження схем і результатів моделювання?
4. Як в програми NI Multisim настроїти колір фону, елементів, проводів та інших позначень схеми?
5. Як в програми NI Multisim настроїти відображення панелей інструментів?

2. СИГНАЛИ ТА МЕТОДИ ЇХ ДОСЛІДЖЕННЯ

2.1. Загальні характеристики сигналів

Форма та способи перетворення електричних сигналів нерозривно пов'язані з принципами побудови електронних пристроїв автоматики та управління і суттєво впливають на їх характеристики та особливості.

Електричні сигнали в електронних пристроях по своїй фізичній суті можна розділити на аналогові та дискретні. Аналогові сигнали є безперервними в часі функціями напруги або струму та, у свою чергу, діляться на постійні і змінні.

Постійні аналогові сигнали є однополярними сигналами напруги або струму, що повільно змінюються з часом.

Змінними аналоговими сигналами називають функції напруги або струму, що змінюються в часі як за амплітудою, так і за знаком. Прикладом змінного сигналу є гармонічний або синусоїдальний сигнал.

Дискретними називаються такі електричні сигнали, які є розривними в часі функціями напруги або струму і можуть мати обмежену кількість рівнів. Найбільш часто в електроніці використовуються дискретні сигнали, які мають тільки два рівні – високої напруги (струму) і низької напруги (струму). Такі сигнали називають імпульсними або двійковими. Представлення інформації за допомогою таких сигналів має ряд переваг, обумовлених високою надійністю та простотою пристроїв, якими вони генеруються і перетворюються. Два дискретні значення, які приймають двійкові сигнали, зазвичай позначають двома цифровими символами – "1« » "0«.»Тому двійкові дискретні сигнали також називають цифровими, а розділ електроніки, що вивчає формування, перетворення та передачу двійкових сигналів – цифровою технікою.

Аналогова або дискретна форми представлення електричних сигналів суттєво впливають на принципи побудови й особливості роботи електронних пристроїв, які формують, підсилю-

ють і перетворюють ці сигнали.

Електронні пристрої, що оперують аналоговими сигналами, як правило, працюють у лінійному режимі та формують клас аналогових пристроїв. Особливість їх полягає в тому, що вхідні й вихідні сигнали зв'язані лінійними або близькими до лінійних залежностями. Прикладами аналогових пристроїв є підсилювачі постійних, змінних та імпульсних сигналів, що працюють без насичення, активні фільтри, генератори гармонічних сигналів, лінійні електричні кола із зосередженими або розподіленими параметрами типу R , L , C , імпульсні трансформатори, що працюють в лінійному режимі без насичення, лінії затримки та ін.

Лінійні елементи використовуються для підсилення, фільтрації, диференціювання, інтегрування, вкорочення, розширення та інших перетворень аналогових і дискретних сигналів.

Електронні пристрої, що оперують дискретними сигналами, працюють у суттєво нелінійному режимі. Основою їх структури є нелінійні (ключові) елементи, які здійснюють під впливом керуючих сигналів різні комутації, підключення та відключення пасивних і активних елементів, джерел живлення та ін. В статичному режимі ключова схема перебуває в одному з двох станів – замкненому (включеному) або розімкненому (виключеному). Комутації ключа створюють на його виході перепади напруги з амплітудою, близькою до амплітуди джерела живлення. Тому на виході ключа формується послідовність імпульсних сигналів, форма яких залежить як від швидкості перемикавання ключа, так і від параметрів лінійних елементів, що входять до складу схеми.

Цифрові пристрої здійснюють логічне перетворення сигналів, їх запам'ятовування, підсумовування; шифрування та дешифрування цифрових кодів; ділення частоти імпульсів, а також ряд інших операцій. Цифрові пристрої відіграють провідну роль у багатьох галузях науки й техніки, і, насамперед в автоматичній, телемеханіці, обчислювальній техніці, супутниковому зв'язку, радіолокації, робототехніці, телебаченні, системах радіо та проводового зв'язку. Це пояснюється тим, що елементи і вузли цифрової техніки завдяки широкому застосуванню в них ключових режимів при існуючому рівні розвитку електроніки є найбільш

надійними, завадостійкими і тому дозволяють забезпечити високу надійність роботи складних апаратних комплексів, наприклад, АСУ, робототехнічних систем, гнучких автоматизованих виробництв, обчислювальних машин та ін.

Важливим фактором, що визначає широке впровадження цифрової техніки, є також її економічна ефективність, яка, з одного боку, визначається технологічністю при виготовленні та простотою при налаштуванні й експлуатації, а з іншого – можливістю розв'язання завдань, нездійсненних раніше на базі вузлів аналогової техніки.

Елементи та вузли цифрової техніки при правильному проектуванні не вимагають індивідуального регулювання й налаштування, дозволяють організувати масове виробництво із застосуванням сучасних засобів автоматизації, скоротити витрати праці й отримати великий економічний ефект. Цифрові пристрої досить просто зазнають автоматизації проектування, для них легко будуються математичні моделі, які з високим ступенем точності відповідають характеристикам реальних пристроїв, вони досить просто перебудовуються (перепрограмовуються) на реалізацію інших функцій. Цифрова техніка дозволяє також широко використовувати мікромініатюризацію, зменшувати масу й габаритні розміри апаратури, що в ряді випадків є одним з вирішальних факторів при проектуванні.

Імпульсна й цифрова техніка, будучи тісно зв'язані одна з одною, відповідають за різні характеристики пристроїв. Імпульсні сигнали є носіями цифрової інформації, при цьому імпульсна техніка займається формуванням, підсиленням і перетворенням імпульсних сигналів за їх фізичними параметрами (тривалістю, частотою, амплітудою, потужністю), а цифрова техніка, абстрагуючись від цих фізичних параметрів, займається перетворенням інформації, яку несуть в собі послідовності імпульсних сигналів, чергування високих і низьких рівнів напруги або сукупності різних рівнів напруги, виражених умовними параметрами 0 або 1, В або А і т. ін. Таким чином, імпульсні сигнали є носіями цифрової інформації, а імпульсні пристрої лежать в основі реалізації будь-яких цифрових схем.

Різна фізична суть імпульсних і цифрових сигналів ви-

значає й різні методи їх дослідження й опису.

В основі аналізу й синтезу імпульсних пристроїв, які є електричними та електронними колами, лежать відомі закони електричних і магнітних кіл, доповнені електричними моделями окремих електронних елементів, що є сукупністю пасивних елементів і джерел напруги. В основі аналізу й синтезу цифрових пристроїв, стан яких описується абстрагованими від конкретних фізичних параметрів змінними й функціями, лежать методи дискретної математики, алгебри логіки й теорії цифрових автоматів.

Інформація, передана за допомогою електричних сигналів, закладена в зміну їх електричних параметрів і форми або в умовному взаємному розташуванні елементів цих сигналів.

При передачі інформації шляхом зміни фізичних параметрів електричних сигналів в якості інформативного параметра можуть використовуватися амплітуда, частота або фаза гармонічного сигналу, а також амплітуда, полярність чи тривалість імпульсу або паузи.

2.2. Аналогові сигнали

Серед аналогових сигналів найбільш поширеними є синусоїдальні сигнали. Математичний вираз, що описує синусоїдальну напругу, має вигляд

$$u = U_m \sin(\omega t + \varphi),$$

де U_m – амплітуда сигналу, ω – кутова частота в радіанах за секунду, причому $\omega = 2\pi f$, φ – початкова фаза сигналу.

Оскільки дослідження характеристик сигналів здійснюється у часовій та частотній областях, то для ілюстрації наведемо представлення сигналів у кожній з цих областей. За допомогою віртуальних інструментів та деяких видів аналізу Multisim можна дослідити характеристики сигналів у часовій області за допомогою осцилографу, а у частотній – за допомогою аналізатора спектру або швидкого перетворення Фур'є.

Змінний електричний сигнал у більшості випадків є функцією часу $x(t)$ з періодом $T = 1/f$, яку можна записати у вигляді ряду Фур'є

$$x(t) = \sum_{n=0}^{N-1} X_n \sin(\omega_n t + \varphi_n),$$

де X_n , ω_n , φ_n – відповідно амплітуда, частота та початкова фаза n -ої гармоніки; N – кількість гармонік.

Окремі складові називають гармоніками, причому коливання основної частоти називають першою гармонікою і т. ін. Сукупність величин X_n називають спектром амплітуд, а сукупність величин φ_n – спектром фаз. Найчастіше цікавляться тільки спектром амплітуд і називають його просто спектром.

На рис. 2.1 наведена схема (2_garm_1.ms10) для дослідження форми і спектру гармонічного сигналу, а на рис. 2.2 – результати аналізу у часовій та частотній областях.

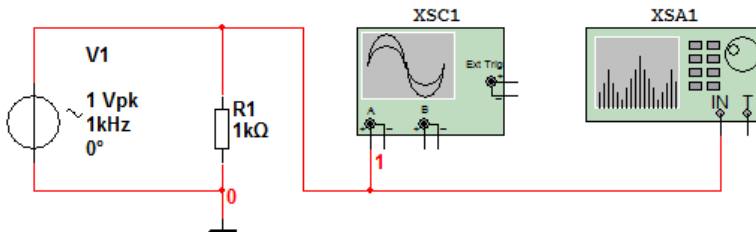
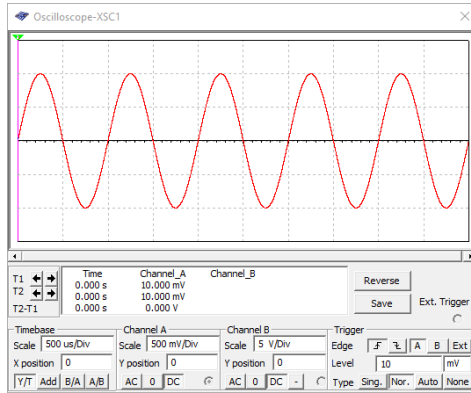


Рис. 2.1. Модель для дослідження форми та спектру гармонічного сигналу

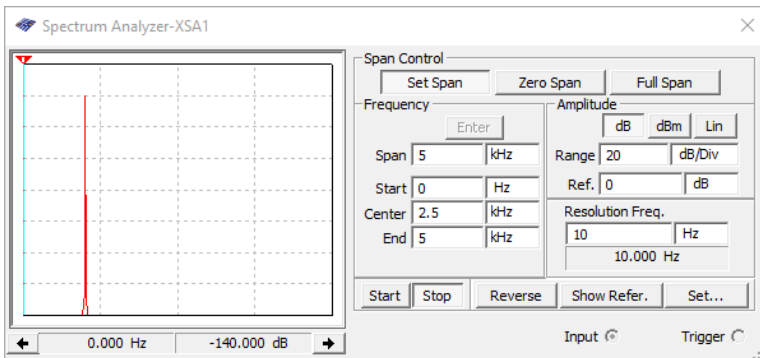
В якості генератора використано генератор гармонічного сигналу V1 (AC voltage). Залишаємо частоту генератора за замовчуванням рівною 1 кГц.

Для формування більш точного значення спектру необхідно встановити обмежений діапазон частот аналізу, достатній для відображення всіх складових спектру. Оскільки гармонічний сигнал у своєму спектрі має містити одну складову з частотою 1 кГц, то достатньо було б встановити верхню частоту діапазону аналізу на рівні 2 кГц. Але оскільки нижче будуть показані спектри більш складних сигналів, то для коректного порівняння результатів доцільно зберегти параметри аналізу однаковими. Тому задаємо верхню частоту смуги аналізу рівною 5 кГц. Форма спектру у значній мірі залежить від розділової здатності аналізу (Resolution Freq. на рис. 2.2, б). Чим меншим є значення розділової здатності, тим точнішим є значення спектру. Тому

задаємо значення розділової здатності 10 Гц. Якщо це значення занадто мале для заданого діапазону частот, то аналізатор спектру автоматично збільшить його до потрібного. Розділова здатність аналізу також залежить від кількості усереднених реалізацій сигналу, яка задається після натискання кнопки Set (рис. 2.2, б) у вікні, що показано на рис. 2.3.



a)



б)

Рис. 2.2. Форма гармонічного сигналу (а) та його спектр (б)

Чим більшою є кількість усереднень, тим більша точність аналізу, але і більший час для його проведення. Значення кількості усереднень може вибиратись у діапазоні від 1024 (2^{10}) до 32768 (2^{15}). Після вибору потрібного числа (у розглянутому прикладі це значення дорівнює 8192 або 2^{13}) треба натиснути кнопку Асепт. Далі можна проводити аналіз, натиснувши кноп-

ку початку моделювання у Multisim.

З рис. 2.2, б слідує, що у спектрі досліджуваного сигналу є тільки одна складова в діапазоні аналізу 140 дБ.

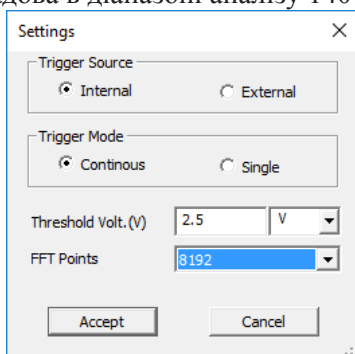


Рис. 2.3. Вікно для вибору кількості усереднених реалізацій сигналу

Інший спосіб дослідження спектру полягає у реалізації швидкого перетворення Фур'є. Для цього потрібно у меню Simulate вибрати пункт Analyses/Fourier Analysis (рис. 2.4) і у вікні, що відкриється (рис. 2.5), задати основну частоту аналізу 1 кГц, кількість гармонік аналізу 5 і масштаб по вертикалі у дБ. Далі необхідно натиснути на вкладку Output і у вікні (рис. 2.6), що відкривається, вибрати точку схеми, у якій проводиться аналіз (на схемі це напруга у точці 1), та натиснути кнопку Simulate. Результати спектрального аналізу наведені на рис. 2.7.

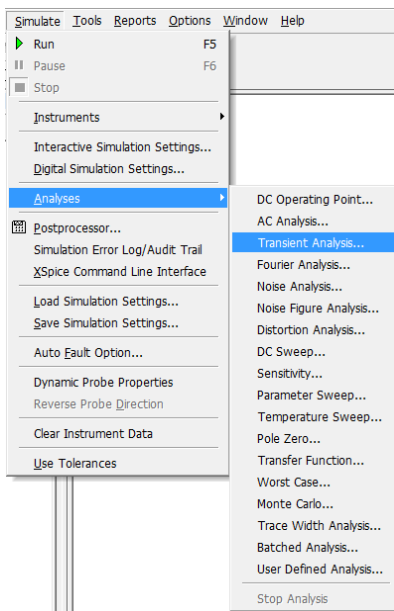


Рис. 2.4. Меню вибору виду аналізу

Окрім значень рівня гармонік у вікні Grapher View (рис. 2.7) виводиться інформація про коефіцієнт гармонік сигналу. Це

параметр THD (Total Harmonic Distortion), який дорівнює 0,0003569% або 3.569 ppm.

Розглянемо, чому найчастіше використовується синусоїдальна функція. Основна її перевага (а також основна причина настільки широкого поширення синусоїдальних сигналів) полягає в тому, що ця функція є розв'язком цілого ряду лінійних диференціальних рівнянь, що описують як фізичні явища, так і властивості лінійних кіл.

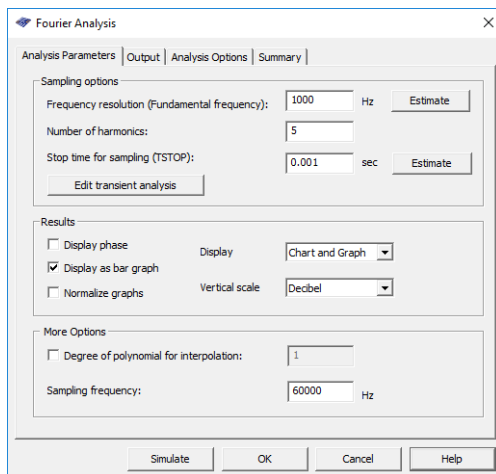


Рис. 2.5. Вікно задавання параметрів Fourier Analysis

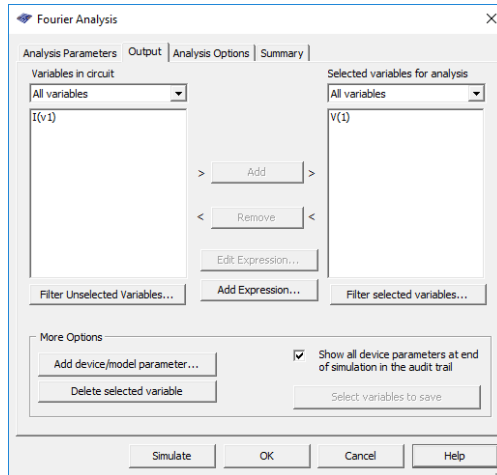


Рис. 2.6. Вікно вибору точки схеми для аналізу

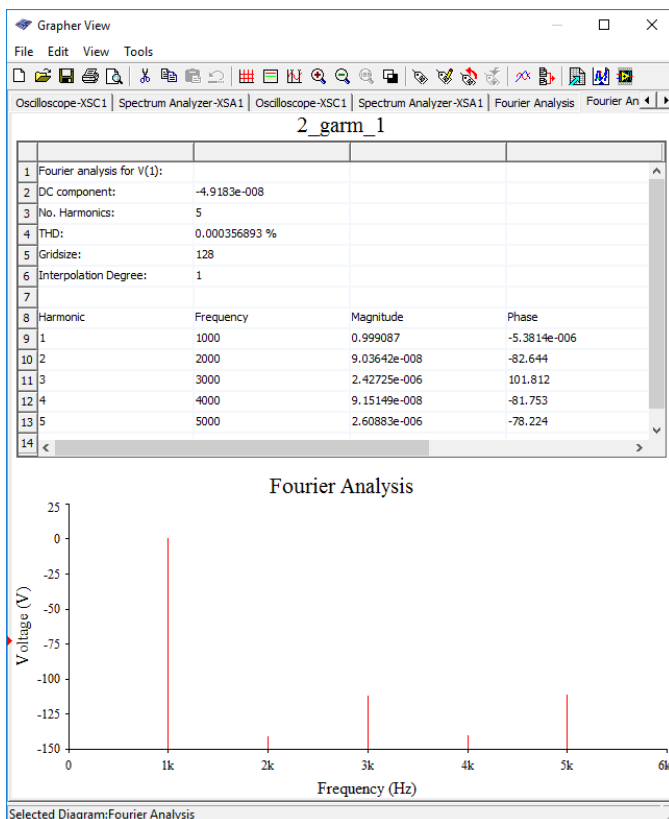


Рис. 2.7. Результати спектрального аналізу гармонічного сигналу, що формується генератором AC Voltage

Лінійне коло має таку властивість: вихідний сигнал, породжений сумою двох вхідних сигналів, дорівнює сумі двох вхідних сигналів, кожний з яких породжений вхідними сигналами, що діють не в сукупності, а окремо: інакше кажучи, якщо $U_{\text{вих}}(A)$ – вихідний сигнал, породжений сигналом A , то для лінійного кола справедливою є така рівність:

$$U_{\text{вих}}(A + B) = U_{\text{вих}}(A) + U_{\text{вих}}(B).$$

Якщо на вході лінійного кола діє синусоїдальний сигнал, то на виході також отримаємо синусоїдальний сигнал, але в загальному випадку його амплітуда і фаза будуть іншими. Це тве-

рдження слухне тільки для синусоїдального сигналу. На практиці прийнято оцінювати реакцію схеми за її амплітудно-частотною характеристикою, яка показує, як змінюється амплітуда синусоїдального сигналу залежно від частоти.

Аналогові сигнали описуються неперервними функціями напруги, струму та ін. Джерелами аналогових сигналів є давачі різних фізичних величин, наприклад, давач тиску, акселерометр та ін.

Процес цілеспрямованої зміни параметрів базового сигналу за законом зміни інформації, що передається, називається модуляцією. Базовий сигнал при цьому називається носійним.

Розрізняють амплітудну (АМ), частотну (ЧМ) та фазову (ФМ) модуляції. Останні дві завдяки взаємозв'язку між частотою та фазою часто об'єднуються під назвою кутової модуляції. Окрім перелічених існує ще багато більш складних видів модуляції. Пристрої, в яких забезпечується процес модуляції параметрів базової функції, називаються модуляторами. Розглянемо коротко основні характеристики двох видів модуляції – амплітудної (АМ) та частотної (ЧМ).

Дослідити форму та спектри АМ- та ЧМ-сигналів можна за допомогою моделі 2_mod.ms10, що наведена на рис. 2.8.

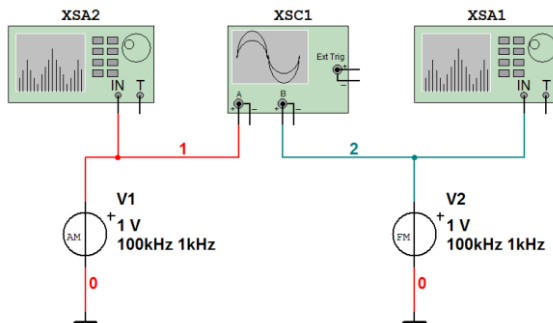
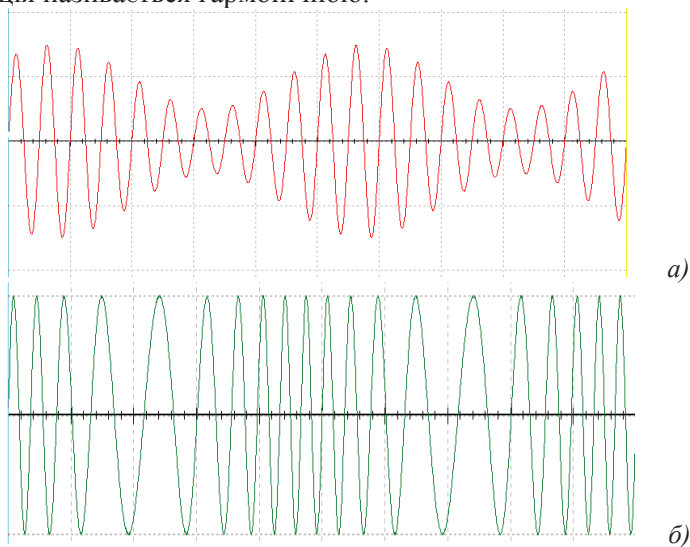


Рис. 2.8. Модель для дослідження форми та спектрів АМ- та ЧМ-сигналів

Для дослідження сигналів вибрана частота носійного коливання 100 кГц, частота модулюючого коливання 1 кГц, коефіцієнт модуляції при АМ дорівнює 0,7, а індекс модуляції при ЧМ – 5.

На рис. 2.9, а наведена форма амплітудно-модульованого (АМ), а на рис. 2.9, б – частотно-модульованого сигналів.

Якщо модулюючий сигнал гармонічний (рис. 2.9), то модуляція називається гармонічною.



*Рис. 2.9. Приклад амплітудно-модульованого (а)
та частотно-модульованого (б) сигналів*

Спектри АМ- та ЧМ-сигналів наведені на рис. 2.10 і 2.11 відповідно. З аналізу отриманих результатів слідує, що у спектрі АМ-сигналу присутні тільки дві бічні смуги частот, а у спектрі ЧМ-сигналу їх значно більше. Теоретично спектр ЧМ-сигналу має нескінченну кількість складових.

Описані види модуляції використовуються для передачі мовних, музичних, телевізійних сигналів, сигналів з різних джерел даних (наприклад, давачів рівня рідини у цистернах, температури в приміщеннях і т. ін.).

Не вдаючись у деталі теорії модуляції, необхідно відзначити, що амплітудна модуляція більш проста в технічній реалізації порівняно з кутовою. Але, у той самий час, цей вид модуляції має низьку стійкість як відносно зовнішніх завад, так і відносно нестабільності параметрів апаратури, в якій вона використовується. Кутова модуляція, навпаки, має більш складну тех-

нічну реалізацію і займає більш широку смугу частот, але при цьому вона більш стійка до зовнішніх завад.

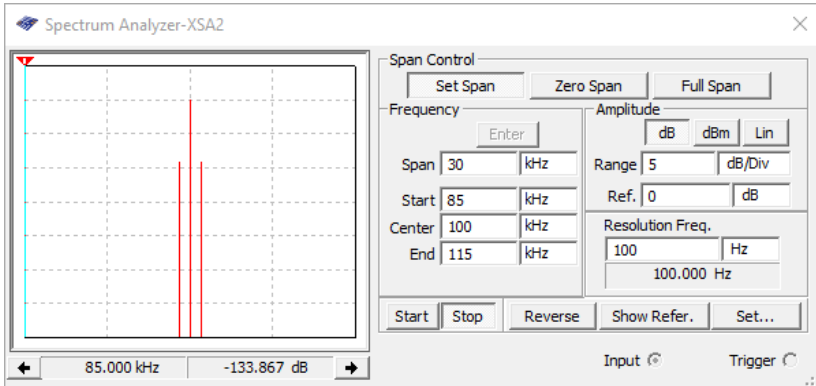


Рис. 2.10. Спектр АМ-сигналу

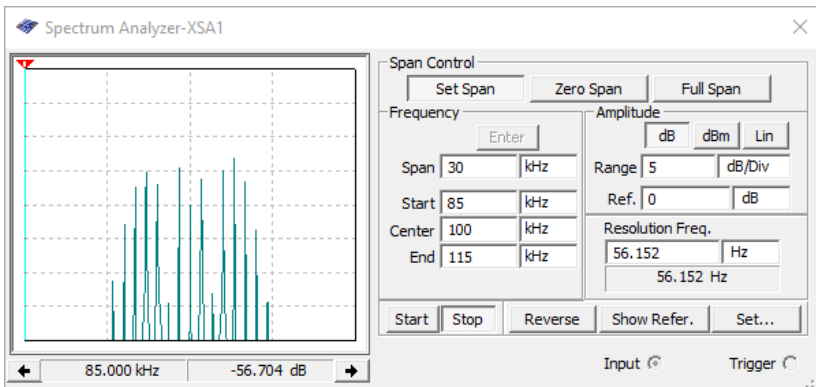


Рис. 2.11. Спектр ЧМ-сигналу

2.3. Імпульсні сигнали

Підтримувати стабільність і точність параметрів аналогових пристроїв досить важко. На їх роботу впливають технологічні допуски, які закладаються при виробництві електронних компонентів, коливання температури, напруги живлення, космічне випромінювання, шуми і наводки, що створюються електронними приладами, а також іншими колами і пристроями.

До того ж, реалізація математичних і логічних операцій у більшості випадків або дуже складна, або навіть неможлива при роботі з аналоговими сигналами. Для того, щоб впевнитись у цьому, достатньо спробувати реалізувати на аналогових компонентах будь-яку з відомих констант. Рекомендується провести такий дослід: використовуючи операційний підсилювач і решту реальних компонентів, реалізувати схему підсилювача так, щоб на виході підтримувалась напруга, яка за рівнем дорівнює числу $\pi = 3,141592653\dots$. На практиці забезпечити точність відтворення такого аналогового сигналу навіть з похибкою в 1% досить складно.

Окрім синусоїдальних в якості базових сигналів часто використовують різні за формою імпульсні сигнали та їх послідовності. На рис. 2.12 зображений типовий поодинокий імпульс та наведені його амплітудні та часові параметри.

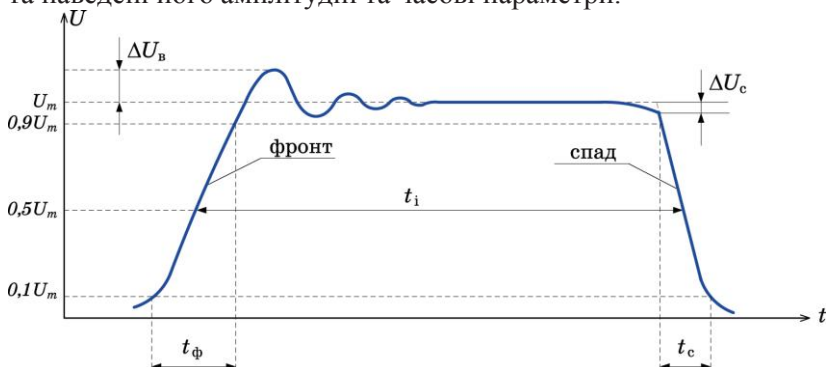


Рис. 2.12. Типовий поодинокий імпульс

До амплітудних параметрів відносяться такі: U_m — амплітуда імпульсу; ΔU_v — викид на фронті імпульсу; ΔU_c — спад вершини. До часових параметрів відносяться тривалість фронту t_ϕ та спаду t_c , а також тривалість імпульсу t_i .

Параметри імпульсу з часом можуть змінюватись у широких межах і, відповідно, поодинокі імпульси можуть мати різну форму. На рис. 2.3, а, б, в зображені імпульси експоненціальної, пилкоподібної та трикутної форм.

Імпульси можуть бути однополярними та різнополярними. Однополярні імпульси можуть бути позитивними та негативними.

вними. Для отримання імпульсних послідовностей різної форми, частоти та амплітуди застосовують спеціальні генератори.

Імпульси, наведені на рис. 2.12 та 2.13, називають відеоімпульсами. Однак окрім відеоімпульсів часто використовуються більш складні імпульси – з заповненням простими або модульованими сигналами. На рис. 2.14 наведені приклади радіоімпульсів з заповненням гармонічним сигналом (рис. 2.14, а) та прямокутним сигналом (рис. 2.14, б).

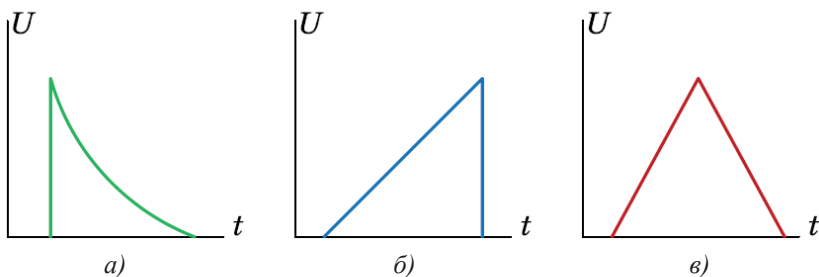


Рис. 2.13. Однополярні імпульси експоненціальної (а), пилкоподібної (б) та трикутної (в) форм

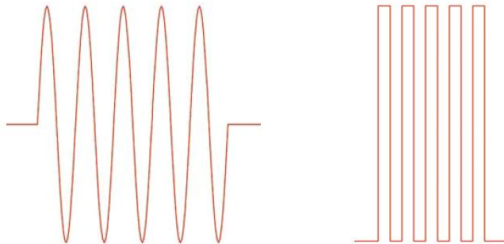


Рис. 2.14. Радіоімпульси з заповненням гармонічним (а) та прямокутним (б) сигналами

Імпульсна послідовність характеризується наявністю пауз між поодинокими імпульсами. Як пауза тривалістю $t_{\text{п}}$ приймається нульовий рівень напруги чи струму або такий рівень, який прирівнюється до нульового (рис. 2.15).

Якщо тривалість імпульсів і пауз між ними в послідовності не змінюється, то говорять про періодичну послідовність, яка характеризується періодом

$$T = t_i + t_{\text{п}},$$

коефіцієнтом заповнення

$$S = t_i / T$$

та шпаруватістю

$$Q = T / t_i.$$

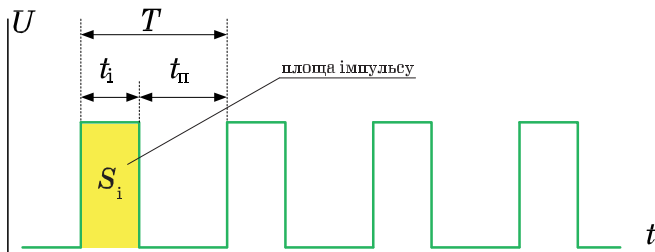


Рис. 2.15. Послідовність прямокутних імпульсів

Одним з найбільш важливих параметрів імпульсної послідовності є середнє значення напруги, яке можна визначити за формулою

$$U_{\text{ср}} = \int_0^{t_i} U(t) dt = \frac{S_i}{T} = \frac{U_m}{Q}, \quad (2.1)$$

де S_i – площа підінтегральної функції; U_m – амплітуда імпульсу.

Графічні зображення спектрів послідовностей деяких імпульсних сигналів наведені на рис. 2.16.

Довжини вертикальних відрізків дорівнюють амплітудам відповідних гармонік. Ці відрізки називають спектральними лініями, а сам спектр – лінійчатим.

В загальному випадку спектр сигналу нескінченний. Тому що амплітуди гармонік при збільшенні їх номера (кутової частоти) зменшуються, починаючи з деякої гармоніки високочастотними коливаннями нехтують, обмежуючи спектр сигналу.

Величина інтервалу частот, в якому розміщується обмежений спектр, називається шириною спектру. Обмеження спектру проводять виходячи з припустимого спотворення сигналу так, щоб не втратити інформацію, яка міститься в ньому.

На рис. 2.17 показано, як формуються прямокутні імпульси (рис. 2.16, а) з окремих гармонік. Графіки першої і третьої гармонік та їх суми зображені на рис. 2.17, а. На рис. 2.17, б ця сума доповнена п'ятою гармонікою. Характерно, що прямокутні

імпульси з шпаруватістю $Q = 2$ є парними відносно осі і містять тільки непарні гармоніки.

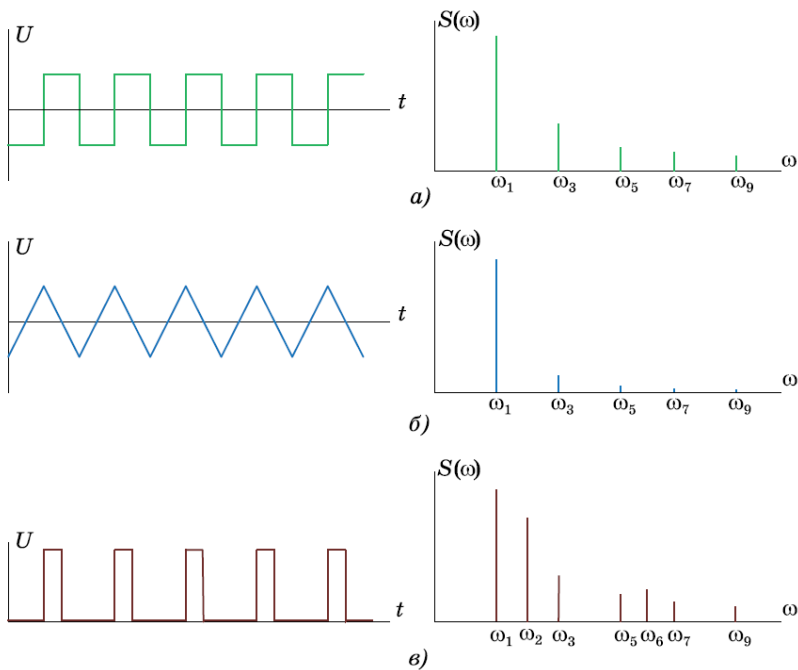


Рис. 2.16. Форма сигналів та графічне зображення спектрів прямокутних симетричних імпульсів з $Q = 2$ (а), трикутних імпульсів (б) та однополярних прямокутних імпульсів з $Q = 4$ (в)

Використання для гармонічного аналізу складних періодичних коливань ряду Фур'є в комбінації з принципом накладання є ефективним засобом для вивчення впливу лінійних кіл на проходження сигналів. Однак необхідно зазначити, що визначення сигналу на виході кола за сумою гармонік із заданими амплітудами та фазами є непростою задачею, особливо якщо не забезпечується швидка збіжність ряду Фур'є який представляє вхідний сигнал. Найбільш поширені в радіотехніці сигнали не відповідають цій умові, тому для задовільного відтворення форми сигналів зазвичай необхідно підсумувати велику кількість гармонік (для прямокутної імпульсної послідовності достатньою є кількість гармонік, що дорівнює потроєному значенню шпару-

ватості).

На рис. 2.18 наведено схему моделі 2_form_imp_z_garm.ms10, що дозволяє формувати сигнали, які складаються з 9 гармонік, а на рис. 2.19 осцилограми послідовності прямокутних імпульсів, сформованих генератором V1 (червона) та послідовності імпульсів з такою ж частотою, сформованої з 9-ти гармонік (зелена). У генератора гармонік моделі (рис. 2.18) встановлені параметри, які відповідають параметрам послідовності прямокутних імпульсів амплітудою 5 В, частотою слідування 1 кГц та коефіцієнтом заповнення 30%.

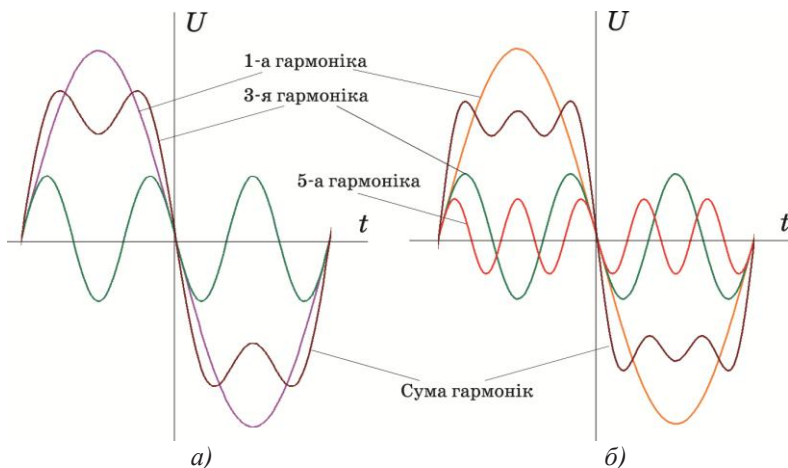


Рис. 2.17. Формування прямокутних імпульсів з першої (фіолетова) та третьої (зелена) гармонік (а), а також з додаванням н'я'ої гармоніки (б)

Як слідує з рис. 2.19, навіть 9-ти гармонік недостатньо для точного відтворення форми сигналу, але отриманий сигнал вже досить близький за формою до сформованого генератором прямокутних імпульсів.

Оскільки при дослідженні різних кіл параметри сигналів можуть змінюватися в широких межах, доцільно мати такий інструмент, який дозволить визначити параметри сигналів з найменшими втратами часу. Найкраще для таких цілей підходить програма NI Multisim.

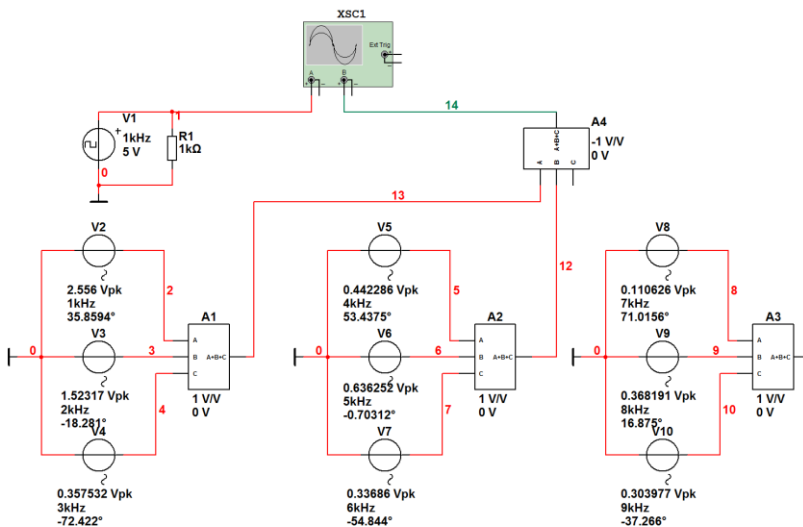


Рис. 2.18. Модель для дослідження форми сигналу, що складається з 9-ти гармонік

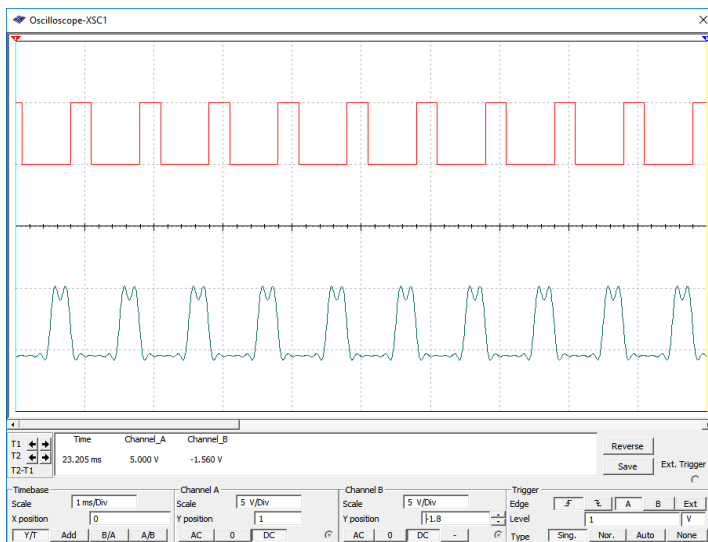


Рис. 2.19. Осцилограми послідовності прямокутних імпульсів, сформованих генератором V1 (червона) та послідовності імпульсів з такою ж частотою, сформованої з 9-ти гармонік (зелена)

2.4. Вимірювання параметрів сигналів за допомогою програми NI Multisim

2.4.1. Джерела сигналів у Multisim

Програма NI Multisim надає широкі можливості як для синтезу сигналів, так і для дослідження їх параметрів. До складу стандартної бібліотеки джерел сигналів входять:

- джерела напруги:

- AC VOLTAGE – джерело змінної напруги;
- AM VOLTAGE – джерело амплітудно-модульованого сигналу;

- BIPOlar VOLTAGE – генератор біполярних прямокутних імпульсів з інтервалами між позитивним та негативним імпульсами;

- CLOCK VOLTAGE – генератор прямокутних однополярних імпульсів;

- EXPONENTIAL VOLTAGE – генератор експоненціального імпульсу;

- FM VOLTAGE – джерело частотно-модульованого сигналу;

- LVM VOLTAGE – генератор LabVIEW, що використовує дані з файлу *.lvm;

- PIECEWISE LINEAR VOLTAGE – генератор кусково-лінійного сигналу;

- PULSE VOLTAGE – генератор біполярних прямокутних імпульсів;

- TDM VOLTAGE – генератор LabVIEW, що використовує дані з файлу *.tdm;

- THERMAL NOISE – генератор теплового шуму;

- TRIANGULAR VOLTAGE – генератор трикутного сигналу;

- джерела струму:

- AC CURRENT – джерело змінного струму;

- BIPOlar CURRENT – генератор струму біполярних

прямокутних імпульсів з інтервалами між позитивним та негативним імпульсами;

- **CLOCK CURRENT** – генератор прямокутних однополярних імпульсів струму;

- **DC CURRENT** – джерело постійного струму;

- **EXPONENTIAL CURRENT** – генератор експоненціального імпульсу струму;

- **FM CURRENT** – джерело частотно-модульованого струму;

- **LVM CURRENT** – генератор LabVIEW, що використовує дані з файлу *.lvm;

- **PIECEWISE LINEAR CURRENT** – генератор кусково-лінійного струму;

- **PULSE CURRENT** – генератор біполярних прямокутних імпульсів струму;

- **TDM CURRENT** – генератор LabVIEW, що використовує дані з файлу *.tdm;

- джерела сигналів, керованих напругою:

- **ABM VOLTAGE** – генератор, параметри якого задаються математичним виразом;

- **CONTROLLED ONE SHOT** – генератор з одним стійким станом (аналог одновібратора), параметри якого задаються користувачем;

- **CURRENT CONTROLLED VOLTAGE SOURCE** – джерело напруги, кероване струмом;

- **FSK VOLTAGE** – джерело частотно-маніпульованого (Frequency Shift Keying) сигналу;

- **VOLTAGE CONTROLLED PIECEWISE LINEAR SOURCE** – джерело напруги, апроксимованої відрізками, кероване напругою;

- **VOLTAGE CONTROLLED SINE WAVE** – джерело синусоїдальної напруги, кероване напругою;

- **VOLTAGE CONTROLLED SQUARE WAVE** – джерело прямокутної напруги, кероване напругою;

- **VOLTAGE CONTROLLED TRIANGLE WAVE** – джерело трикутної напруги, кероване напругою;

- **VOLTAGE CONTROLLED VOLTAGE SOURCE** – джерело напруги, кероване напругою;
- джерела сигналів, керовані струмом;
- **ABM CURRENT** – джерело струму;
- **CURRENT CONTROLLED CURRENT SOURCE** – джерело струму, кероване струмом;
- **VOLTAGE CONTROLLED CURRENT SOURCE** – джерело струму, кероване напругою.

2.4.2. Інструменти аналізу сигналів у Multisim

У Multisim передбачена велика кількість функцій аналізу даних емуляції – від простих до самих складних, у тому числі й вкладених.

Список усіх можливих видів аналізу Multisim наведений на рис. 2.20. Розглянемо основні види аналізу.

1. DC Operating Point – аналіз кола на постійному струмі. Аналіз кіл на постійному струмі здійснюється для резистивних схем. Це правило впливає прямо з теорії електричних кіл. При аналізі на постійному струмі конденсатори замінюють розривом, котушки індуктивності – коротким замиканням, а нелінійні компоненти, такі як діоди і транзистори, замінюють їх опором постійному струму в робочій точці. Аналіз кіл на постійному струмі виявляє вузлові потенціали досліджуваної схеми.

2. AC Analysis – аналіз кола на змінному струмі. Такий аналіз полягає в побудові частотних характеристик.

3. Transient Analysis – аналіз перехідних процесів. Дозволяє визначити форму вихідного сигналу протягом тривалості перехідного процесу, тобто побудувати графік сигналу в функції часу. Для того, щоб почати аналіз, потрібно вибрати пункт меню Simulate/Analyses та необхідний режим.

4. Fourier Analysis – Фур'є-аналіз, тобто аналіз спектру сигналу. Дозволяє здійснювати аналіз спектру сигналів у заданому діапазоні частот і вибраній кількості гармонік. Результати аналізу виводяться як у вигляді графіків, так і в табличній формі.

З іншими методами аналізу за необхідністю будемо зна-

йомитись в процесі досліджень.

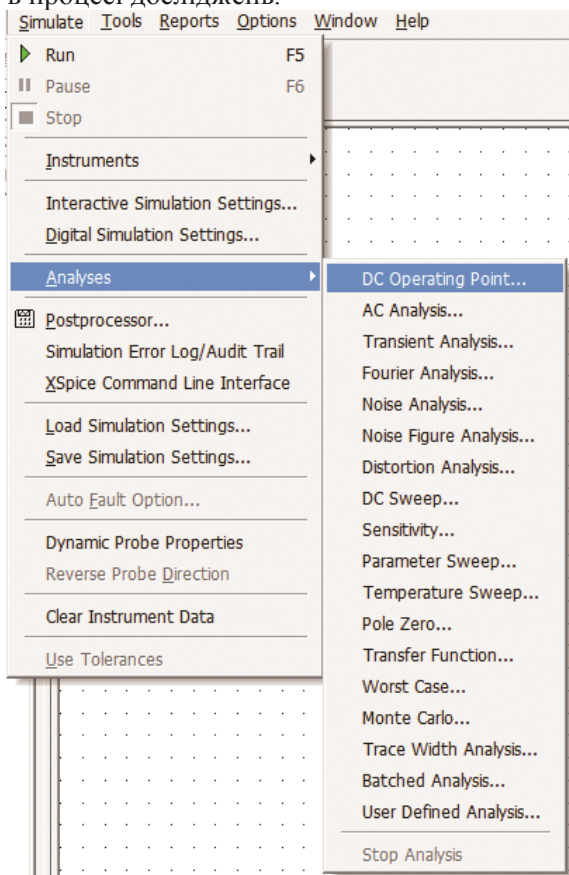


Рис. 2.20. Меню аналізу програми NI Multisim

Крім вбудованих функцій аналізу є можливість визначити власну необхідну функцію за допомогою команд SPICE.

При підготовці до аналізу необхідно налаштувати його параметри, наприклад, діапазон частот для аналізу на змінному струмі (AC analysis) або основну частоту та кількість гармонік для Фур'є аналізу. Необхідно також вибрати вихідні канали (traces), що позначені на схемі як ноди, тобто точки (вузли) схеми, для яких необхідно провести аналіз.

Для перегляду результатів емуляції призначено вікно

Grapher. Воно відкривається з меню View/Grapher або іконкою, що розташована у верхньому рядку графічного меню програми. Після завершення вибраного типу аналізу вікно Grapher відкривається автоматично.

Більшість параметрів Grapher настраюється у вікні властивостей. Наприклад, можна змінювати масштаби, діапазони, заголовки, стилі ліній осей та ін. (рис. 2.21).

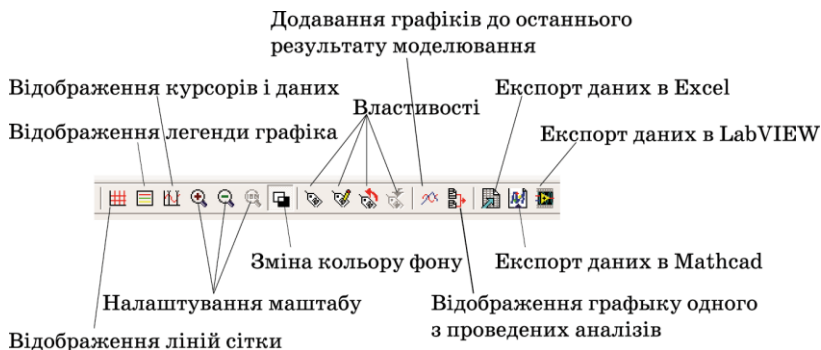


Рис. 2.21. Меню настроювання параметрів Grapher

Кнопками управління (рис. 2.21), піктограми яких відображаються у верхній частині вікна, можна виконувати такі дії.

1. Включити або виключити сітку.
2. Включити або виключити легенду (позначення кольором сигналу у вибраному каналі вимірювання).
3. Включити або виключити курсори. При цьому користувачу надається можливість переміщення курсорів по горизонтальній осі. У вікні, яке відкривається знизу під вікном графіків, відображаються:

- x_1 – відстань першого курсору (блакитного кольору) від початку координат в одиницях часу;
- y_1 – значення напруги в точці перетину сигналу з курсором. Аналогічно для другого (жовтого) курсору. Крім того, вказуються мінімальні та максимальні значення сигналів і зсув по осях x та y .

4. Змінити колір фону (з чорного на білий і навпаки) кнопкою реверс (Reverse).

5. Змінити оформлення вікна перегляду. При натисканні

кнопки Graph Properties відкривається вікно настроювання (рис. 2.22), у якому в кожній вкладці можна внести зміни в оформлення.

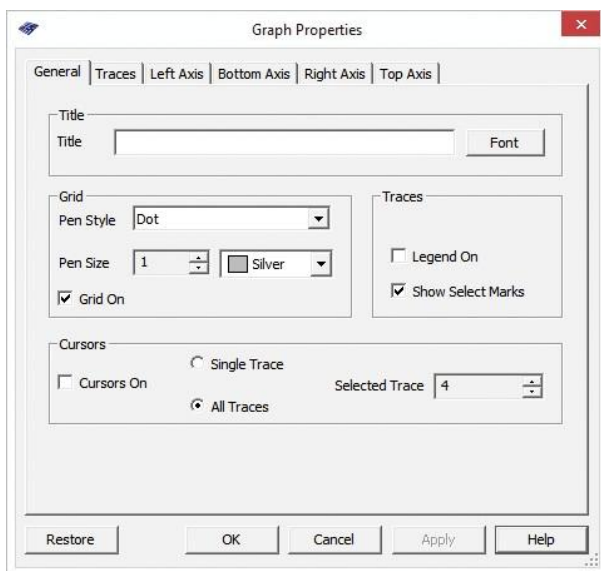


Рис. 2.22. Вікно настроювання інтерфейсу Graph Properties

На вкладці General задається колір і товщина ліній сітки, число виведених кривих. На ній також можна включити і виключити курсори, сітку та легенду.

На вкладці Traces вибирається колір і товщина ліній виведених кривих, зсув графіка по осях (вручну або автоматично), на осі Left Axis – мінімальне та максимальне значення напруги (діапазон вимірювання напруги), напис на осі (Label), колір і товщина осі, кількість поділок, кількість знаків після коми в числах, нанесених на осі, а також тип шрифту.

Кнопки масштабування, збереження та завантаження настроювань, додавання наступних результатів вимірювання в особливих поясненнях не мають потреби.

Натиснувши кнопку Overlay traces, можна додати вікно з результатами вимірювань з двох різних вікон на одному графіку. В таблицю, що відображає значення напруги у різних точках осцилограм та інші параметри (наприклад, виміряні при інших

параметрах і виведені в інше вікно), виводяться дані двох графіків. Біля назви додаткових стовпців виведеної таблиці (рис. 2.23) з'являється напис (merged). На рис. 2.23 до результатів вимірювання спектру у вузлі V(2) додано результати вимірювання у вузлі V(1).

Fourier Analysis		
	V (2)	V (1) [Merged]
x1	2.0000k	2.0000k
y1	1.1794e-016	636.6822m
x2	1000.0000	1000.0000
y2	1.2733	900.3369m
dx	-1000.0000	-1000.0000
dy	1.2733	263.6546m
1/dx	-1.0000m	-1.0000m
1/dy	785.3802m	3.7928
min x	0.0000	0.0000
max x	10.0000k	10.0000k
min y	0.0000	0.0000
max y	1.2733	900.3389m
offset x	0.0000	0.0000
offset y	0.0000	0.0000

Рис. 2.23. Вікно результатів вимірювання після натискання кнопки *Overlay traces*

Для трансформації часового масштабу відображуваної осцилограми або АЧХ необхідно, натиснувши ліву кнопку "м«-ші",» виділити ділянку сигналу (така операція можлива тільки у вікні, в якому відображаються сигнали, обмежені осями) і відпустити кнопку. У виділеній ділянці масштаб зображення по вертикалі залишиться незмінним, а по горизонталі він розтягнеться на все вікно. Чим менше ширина захопленої ділянки, тим більшим є коефіцієнт збільшення масштабу зображення по горизонталі.

Останні три кнопки призначені для експорту результатів вимірювань в Excel, MathCad і збереження файлу вимірювань (Save to Measurement file) для системи Lab View.

Програма пакету Multisim Postprocessor за допомогою Grapher дозволяє відобразити результати моделювання в графічному вигляді.

Для роботи з функцією Postprocessor необхідно знати назви вузлів. Тільки ті параметри (вхідні та вихідні змінні), які

вказуються при виконанні будь-якого виду аналізу (AC Sweep, DC Sweep, Transient Analysis та ін.) відображаються на графіках Postprocessor і Grapher.

За допомогою даної функції можна створити декілька графіків, змінювати параметри графіка, видаляти об'єкти, проводити логічні та алгебраїчні операції над графіками (додавання, множення, піднесення до квадрату та ін.).

2.4.3. Налаштування параметрів аналізу

Перш ніж проводити будь-який аналіз, необхідно встановити його вихідні параметри, задавши їх у пунктах меню Interactive Simulation Settings та Digital Simulation Settings. Якщо ці установки не провести, моделювання буде виконуватися з параметрами, заданими в програмі Multisim за замовчуванням.

При зверненні до пункту меню Interactive Simulation Settings відкриється вкладка Defaults for Transient Analysis Instruments, у якій задаються початковий і кінцевий час аналізу. За замовчуванням – від 0 до 1030 с, при цьому за замовчуванням часовий крок аналізу задається автоматично. Однак досить поставити "гачок" у пункту Set Initial Time Step, щоб встановлювати крок аналізу вручну.

В закладці **Analysis Options** встановлені за замовчуванням такі параметри:

- Use Multisim Defaults (параметри за замовчуванням);
- Perform Consistency check before starting analysis (перевірка правильності зв'язків до початку аналізу);
- Discard plots to save memory (не проводити запис графіків).

Для того, щоб змінити параметри, задані за замовчуванням, необхідно вибрати режим Use Custom Settings і натиснути кнопку Customize, при цьому відкриється вікно Custom Analysis Options (рис. 2.24) – набір команд для встановлення параметрів моделювання.

Налаштування загального характеру (**Global**) задаються в діалоговому вікні (рис. 2.24), у якому можна змінити параметри аналізу, якщо поставити "гачок" у відповідного пункту.

Параметри мають такі значення:

Parameter	ON	Value	Unit
Absolute error tolerance [ABSTOL]	<input type="checkbox"/>	1e-012	Amperes
Voltage error tolerance [VNTOL]	<input type="checkbox"/>	1e-006	Volts
Charge error tolerance [CHGTOL]	<input type="checkbox"/>	1e-014	Coulombs
Relative error tolerance [RELTOL]	<input type="checkbox"/>	0.001	
Minimum conductance [GMIN]	<input type="checkbox"/>	1e-012	Mho
Minimum acceptable ratio of pivot [PIVREL]	<input type="checkbox"/>	0.001	
Minimum acceptable pivot [PIVTOL]	<input type="checkbox"/>	1e-013	
Operating temperature [TEMP]	<input type="checkbox"/>	27	°C
Shunt resistance from analog nodes to ground [RSHUNT]	<input checked="" type="checkbox"/>	1e+012	Ω
Transient analysis supply ramping time [RAMPTIME]	<input type="checkbox"/>	0	Seconds
Fractional step allowed by code model inputs between iterations [CONVSTEP]	<input type="checkbox"/>	0.25	
Absolute step allowed by code model inputs between iterations [CONVABSTEP]	<input type="checkbox"/>	0.1	
Enable convergence assistance for code models [CONVLIMIT]	<input checked="" type="checkbox"/>		
Print simulation statistics [ACCT]	<input type="checkbox"/>		

Restore to Recommended Settings

OK Cancel Help

Рис. 2.24. Вікно установки глобальних параметрів аналізу

- ABSTOL – абсолютна похибка розрахунку струмів; якщо струми у схемі моделі значно більші зазначеного на рис. 2.24 значення, то для прискорення процесу аналізу значення ABSTOL доцільно збільшити, виходячи з припустимої похибки розрахунків (наприклад, з врахуванням розрядності мультиметра);

- VNTOL – абсолютна похибка розрахунку напруги;

- CHGTOL – абсолютна похибка розрахунку зарядів; встановлені за замовчуванням значення (рис. 2.24) змінювати не рекомендується;

- GMIN – мінімальна провідність вітки або ділянки кола (зазначене на рис. 2.24 значення змінювати не рекомендується); збільшення GMIN позитивно позначається на сходженні розв'язку при одночасному зниженні точності моделювання; використовується при підключенні додаткового алгоритму Gmin stepping;

- PIVREL, PIVTOL – відносна та абсолютна величини елементу рядка матриці вузлових провідностей, прийнятого за основний; введення такого елемента у випадку застосування методу Гауса дозволяє підвищити точність проміжних обчислень і, отже, зменшити загальну кількість ітерацій; встановлені за замовчуванням значення параметрів (рис. 2.24) змінювати не рекомендується;

- RELTOL – відносна помилка моделювання; впливає на збіжність розв'язку та швидкість моделювання; рекомендовані значення – $10^{-2} \dots 10^{-6}$;

- TEMP – температура, для якої проводиться моделювання;

- RSHUNT – опір витоку для всіх нод (точок) схеми відносно спільного проводу (шини заземлення); при повідомленнях про помилки "N« DC path to ground" »погане заземлення кола) і "M« trix is nearly singular" »матриці дуже близькі) мова йде про матриці Li U-рівнів) значення параметра варіюється в межах $10^6 \dots 10^9$ Ом;

- RAMPTIME – початкова точка відліку часу при аналізі перехідних процесів;

- CONVSTEP, CONVABSSTEP відносний і абсолютний розміри автоматично контрольованого кроку ітерації при розрахунку режиму по постійному струмі;

- CONVLIMIT – вмикання або вимикання додаткових засобів, вбудованих у моделі деяких компонентів, для забезпечення збіжності ітераційного процесу;

- ACCT – вкл./викл. виводу отриманих в результаті моделювання статистичних даних, які можуть бути корисними для налаштування процесу моделювання при виникненні проблем (дані виводяться в діалоговому вікні Grapher).

Усі рекомендовані за замовчуванням параметри наведені в підказці до таблиці Global tab програми. Кнопка Restore to Recommended Settings у всіх діалогових вікнах призначена для відновлення значень параметрів аналізу, встановлених за замовчуванням. Використовується в тому випадку, якщо після редагування необхідно повернутися до даних за замовчуванням.

Слід враховувати, що результатом зміни параметрів

ABSTOL, RELTOL, VNTOL, CHGTOL, наприклад, з метою зменшення часу моделювання, може стати незадовільна збіжність ітераційного процесу при наявності в схемі вузлів і кіл, що мають близькі значення шуканих значень напруги, струму або заряду.

На вкладці вікна установки параметрів **DC-аналізу** проводяться налаштування для розрахунку режиму за постійним струмом (статичний режим) за допомогою діалогового вікна, в якому задаються такі параметри (рис. 2.25):

- ITL1 – максимальна кількість ітерацій; при повідомленні "N« convergence in DC analysis" »незадовільна збіжність при DC-аналізі) значення параметра необхідно збільшити до 500...1000;

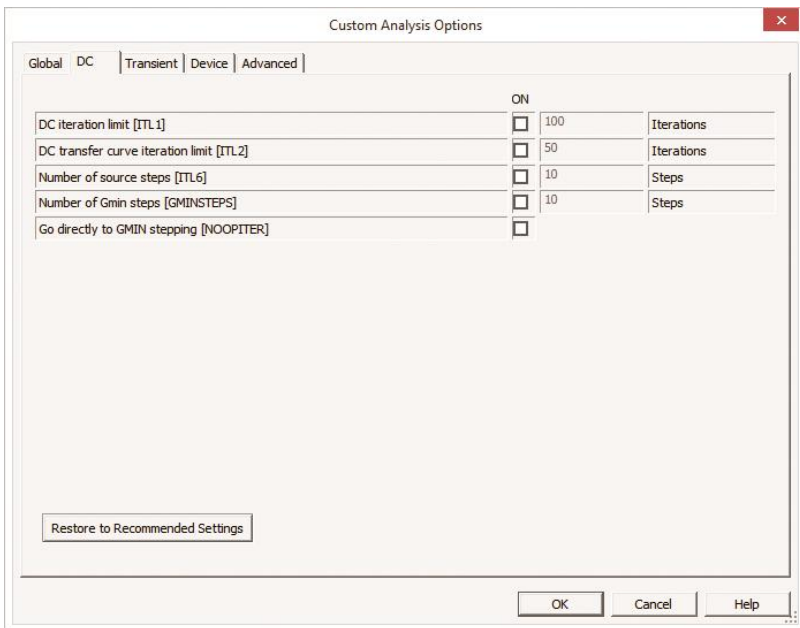


Рис. 2.25. Вікно установки параметрів аналізу на постійному струмі

- ITL2 – максимальна кількість кроків додаткового алгоритму Source stepping, що використовується при незадовільній збіжності розв'язку для нелінійних кіл. При виконанні цього алгоритму здійснюється "р«зподіл" »ілянки характеристики нелі-

нійного компонента на задану параметром SRCSTEPS ("Steps in source stepping algorithm")» кількість кроків;

- ITL6 – кількість кроків додаткового Gmin stepping алгоритму, який використовується при незадовільній збіжності;

- GMINSTEPS – кількість ітерацій з кроком Gmin (мінімальна провідність кола) додаткового Gmin stepping алгоритму;

- NOOPITER – перехід безпосередньо до покрокового виконання алгоритму Gmin.

На вкладці вікна (рис. 2.26) встановлення параметрів режиму аналізу перехідних процесів (**Transient**) задаються такі параметри:

- ITL4 – максимальна кількість ітерацій для розрахунку однієї точки перехідного процесу; при повідомленні "Time step too small" (крок часу дуже малий) або "Non convergence in the transient analysis" (немає сходження) значення параметра доцільно збільшити до 15 і повторити аналіз;

- Maxcorder – максимальний порядок інтегрування – параметр, який необхідно задати для наближеного інтегрування системи диференціальних рівнянь методом Гіра (GEAR), що передбачає можливість роботи зі змінним кроком, який, залежно від швидкості зміни змінної (потенціалу V_i), може автоматично змінюватися; при збільшенні порядку точність розрахунків збільшується з одночасним зменшенням швидкості процесу моделювання; за замовчуванням параметр дорівнює 2, діапазон зміни від 2 до 6;

- TRTOL – константа, що визначає точність розрахунків; наприклад, у найпростіших випадках така константа, що називається характеристикою кроку, визначається співвідношенням $|dV/dt| h = \text{const}$, де h – крок ітерації, тобто при зміні $|dV/dt|$ має відповідно мінятися і крок ітерації; встановлене за замовчуванням значення TRTOL дорівнює 7, а змінювати його не рекомендується;

- METHOD – метод наближеного інтегрування системи диференціальних рівнянь TRAPEZOIDAL (метод Ейлера з вирівнюванням), що реалізує формулу

$$V_{n+1} = V_n + (h/2)(dV_n/dt + dV_{n+1}/dt),$$

де h – крок ітерації, V_{n+1} та V_n – значення потенціалу в i -й точці

схеми на поточному та попередньому кроках ітерації.

Цей метод використовується, якщо в процесі моделювання виникають небажані коливання числових значень розрахованих параметрів або у випадку використання в схемі ідеальних перемикачів. За замовчуванням цей метод використовується для аналізу схем генераторів.

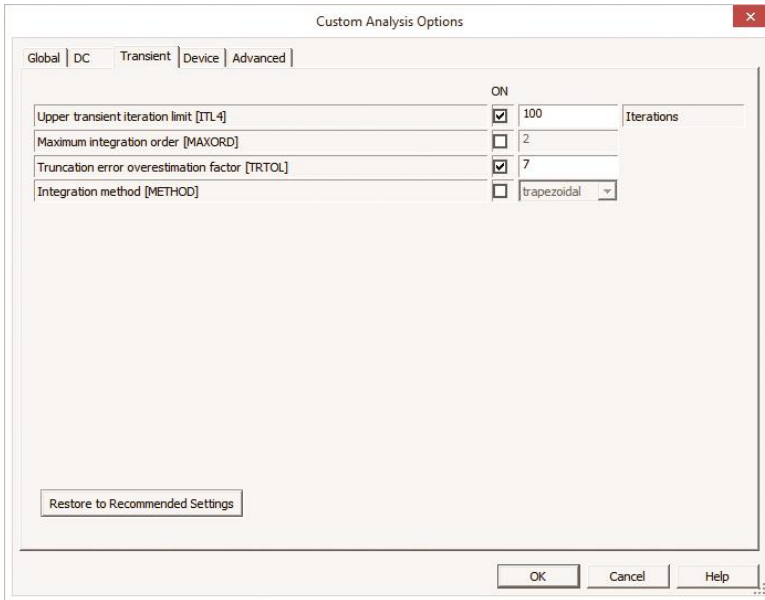


Рис. 2.26. Вікно установки параметрів аналізу перехідних процесів (Transient)

На вкладці вікна (рис. 2.27) установки параметрів **Device** (МОН-транзисторів) задаються такі параметри:

- TNOM – номінальна температура компонента;
- BYPASS – вмикання або вимикання нелінійної частини моделі компонента;
- DEFAD – площа дифузійної області стоку, м^2 ;
- DEFAS – площа дифузійної області джерела, м^2 ;
- DEFL – довжина каналу польового транзистора, м;
- DEFW – ширина каналу, м;
- TRYTOCOMPACT – вмикання або вимикання лінійної частини моделі компонента;

- **OLDLIMIT** – вмикання або вимикання обмежень SPICE2-моделі для МОН-транзисторів (MOSFET), за замовчуванням виключене.

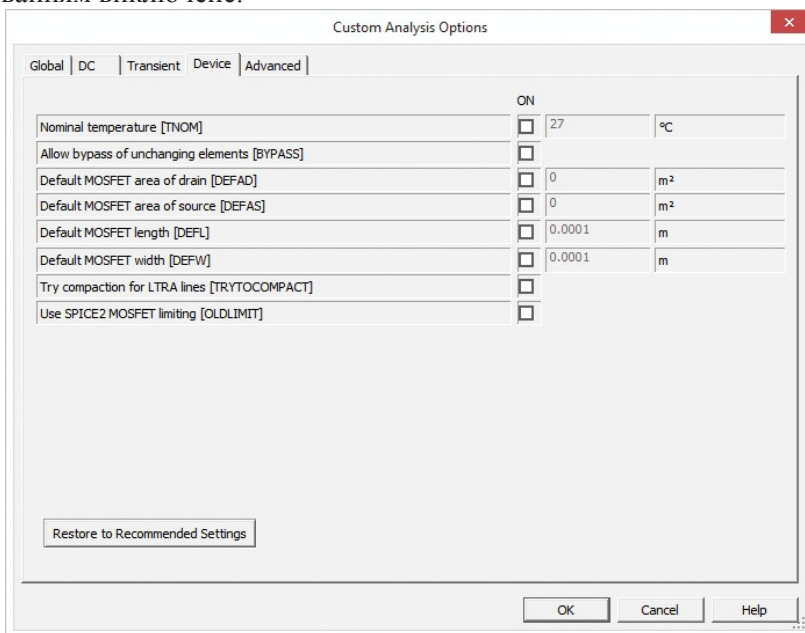


Рис. 2.27. Вікно установки параметрів аналізу Device (МОН-транзисторів)

Остання вкладка в цьому пункті меню – **Advanced** (рис. 2.28), на якій можна встановити такі параметри:

- **AUTOPARTIAL** – використання розрахунку auto-partial для всіх моделей;
- **BADMOS3** – використання старої моделі mos3;
- **KEEPOPINFO** – запис кожної точки при обчисленні параметрів малого сигналу;
- **MAXEVTITER** – максимальна кількість ітерацій в точці аналізу;
- **MAXOPALTER** – максимальна кількість ітерацій в точці аналізу (альтернативний метод) для аналізу кіл за постійним струмом (DCOP);
- **MINBREAK** – мінімальний часовий інтервал між точ-

ками переривання;

- NOOPALTER – не використовувати альтернативний метод для аналізу кіл за постійним струмом (DCOP).

Більш докладну інформацію про настроювання пунктів меню "С«stom Analyzes Option" »ожна отримати в [4], що входить до складу документації до програми і розміщено за замовчуванням у папці C:\Program Files\National Instruments\Circuit Design Suite 10.0\documentation\Multisim 10 User Guide.pdf.

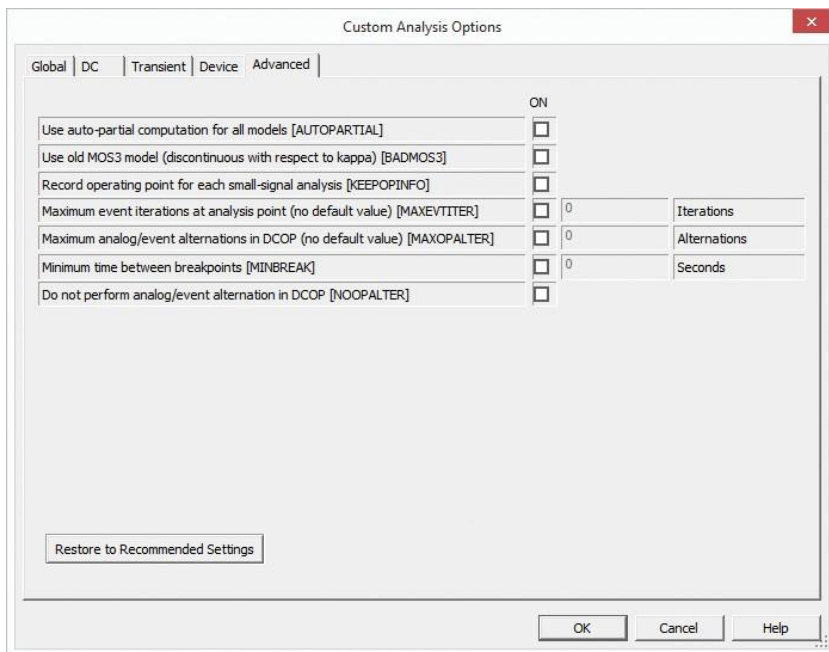


Рис. 2.28. Вікно установки параметрів аналізу Advanced

2.5. Основні правила роботи з вимірювальними приладами

2.5.1. Осцилограф

В програмі Multisim доступно 4 типи осцилографів. Розглянемо особливості роботи з дво- і чотириканальним осцилог-

рафами. Органи керування двоканальним осцилографом показані на рис. 2.29.

Масштаб зображення по вертикалі регулюється дискретно і незалежно в кожному каналі органами регулювання чутливості входу. Масштаб зображення по горизонталі регулюється тривалістю розгортки. В кожному каналі передбачена можливість зсуву зображення по вертикалі як вгору, так і вниз. Зсув по горизонталі здійснюється за допомогою зміни значення "X«position"». Для того, щоб зображення не зміщувалося по горизонталі, необхідно синхронізувати розгортку осцилографа з початком наступного періоду сигналу. Для синхронізації зазвичай вибирають режим "N« κ " » задають рівень синхронізації в рядку "L« κ vel", який дорівнює від 10% до 50% амплітуди сигналу, яким здійснюється синхронізація. Це може бути сигнал каналів А або В (за фронтом або спадом), а також сигнал зовнішнього джерела сигналу, що підключається до входу "«xt Trg"». При дослідженні сигналів різної частоти слід вибирати для синхронізації сигнал з мінімальною частотою.

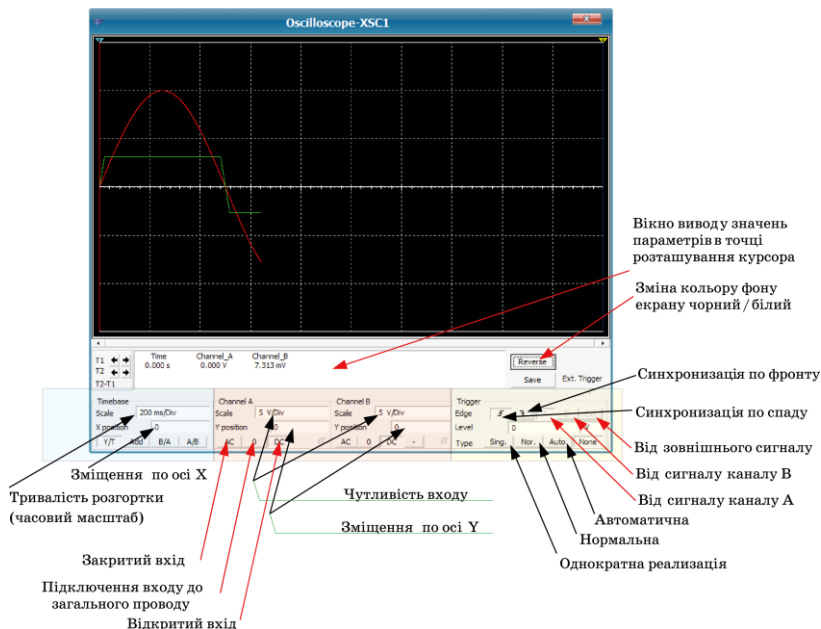


Рис. 2.29. Органи керування двоканальним осцилографом

У чотириканальному осцилографі замість роздільних вікон регулювання чутливості і зміщення по вертикалі використовується одне вікно і перемикач каналів (рис. 2.30). Для регулювання в потрібному каналі необхідно встановити покажчик (біла рисочка) на необхідну літеру і задати параметри у вибраному каналі. Аналогічно здійснюється регулювання і в інших каналах.

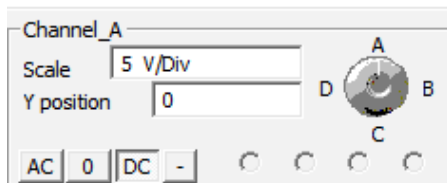


Рис. 2.30. Органи керування чутливістю чотириканального осцилографа

Вимірювання параметрів сигналу в будь якій точці (наприклад, амплітуди і часового інтервалу від початку розгортки) можна здійснювати як за поділками, нанесеним на екрані, так і за допомогою курсорів (зелений і жовтий трикутники у верхній частині екрана осцилографа). Досить перемістити курсор в потрібну точку екрану і у вікні виводу значень параметрів в точці розміщення курсора прочитати покази (рис. 2.31).

Як впливає з рис. 2.31, напруга в точці, поміченій червоним курсором, складає величину 994,954 мВ для каналу А і 2,487 В для каналу В, а в точці, поміченій синім курсором – відповідно 999,436 мВ і 2,499 В. У нижньому рядку вказується часовий інтервал між курсорами і різниця напруг, виміряних в цих точках.

2.5.2. Плотер Боді

Для того, щоб використати даний інструмент, необхідно натиснути на кнопку Bode Plotter на панелі інструментів, і перемістити курсор на місце, де потрібно розмістити іконку на робочій області. Іконка з умовним позначенням плотера (рис. 2.32, а) використовується для його підключення до схеми.

Плотер використовується для побудови амплітудно- і

фазочастотних характеристик (АЧХ і ФЧХ). Коли плотер підключається до схеми, виконується спектральний аналіз у заданому діапазоні частот. Для роботи Бодє-плотера на вході схеми обов'язково потрібно включити джерело сигналів змінного струму (АС), як показано на рис. 2.32, б. Спільний провід можна до плотера не підключати, оскільки за замовчуванням це з'єднання здійснюється автоматично, хоча на схемі це і не відображається. Частота сигналу джерела АС не впливає на результати роботи плотера і може бути встановлена будь-якою.

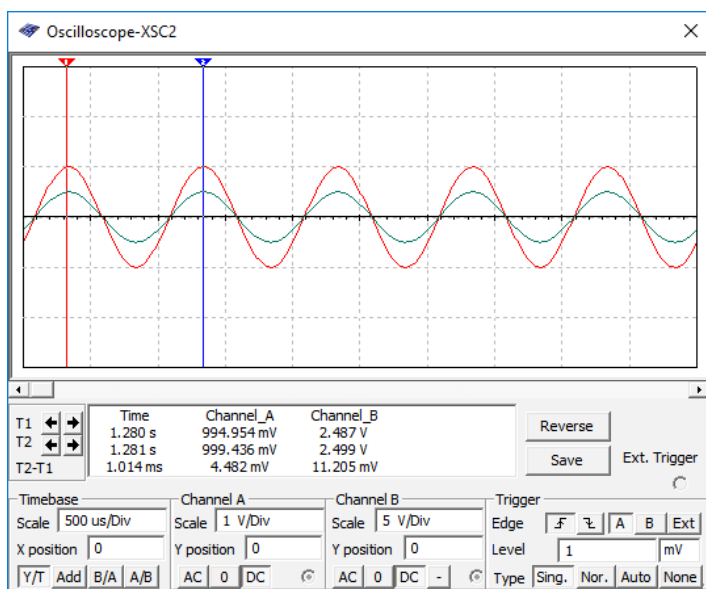


Рис. 2.31. Вимірювання параметрів сигналу в точці, позначеній курсором

Подвійне клацання по іконці відкриває панель інструменту, яка використовується для вводу установок і перегляду результатів вимірювань (рис. 2.33).

Початкове і кінцеве значення масштабу по вертикалі та горизонталі встановлені за замовченням на максимум. Ці значення можуть змінюватися для перегляду графіка в різних масштабах. Якщо масштаб збільшувався або змінювалась база після закінчення симуляції, вам може знадобитись повторна активація

схеми, щоб отримати більш детальне зображення. Якщо виводи плотера Бодє переносяться до інших вузлів, необхідно повторно активувати схему, щоб отримати правильні результати.

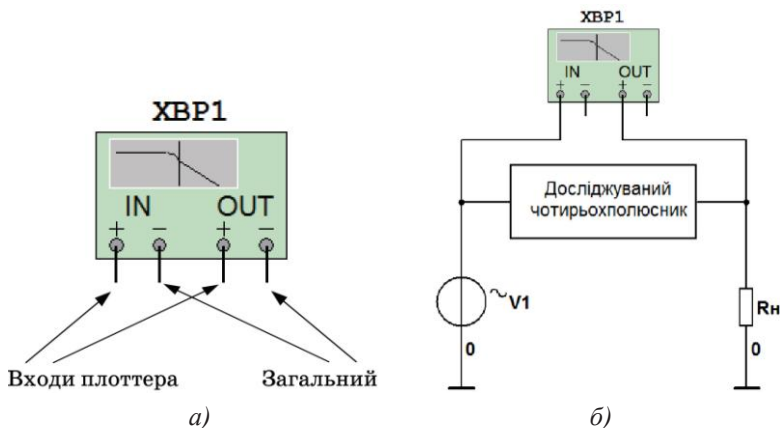


Рис. 2.32. Призначення входів (а) та схема підключення (б) плотера Бодє

Для налаштування кількості точок, що відображають АЧХ або ФЧХ, необхідно натиснути кнопку Set (рис. 2.33) і у вікні Settings dialog, що відкривається, встановити потрібну кількість точок (Resolution Points) і натиснути кнопку Accept. Максимальне значення кількості точок дорівнює 1000. Бажано завжди використовувати максимальне значення для підвищення точності отриманих результатів.

Вимірювання АЧХ та ФЧХ (Magnitude або Phase).

В режимі вимірювання АЧХ (Magnitude) вимірюється відношення величин напруги між двома вузлами в схемі, які під'єнані до входів In та Out. В режимі вимірювання ФЧХ між цими вузлами вимірюється зсув фаз.

Установки масштабу вертикальної та горизонтальної осей.

Логарифмічна шкала використовується, коли значення, які порівнюються, значно відрізняються за рівнем.

Налаштування частотного діапазону аналізу здійснюється у вікні Horizontal шляхом установки верхнього значення частоти (вікно F на рис. 2.34) і нижнього значення частоти (вікно I

на рис. 2.34).

Настроювання максимального і мінімального рівнів вхідного сигналу здійснюється у вікні Vertical (відповідно вікна F та I на рис. 2.34).

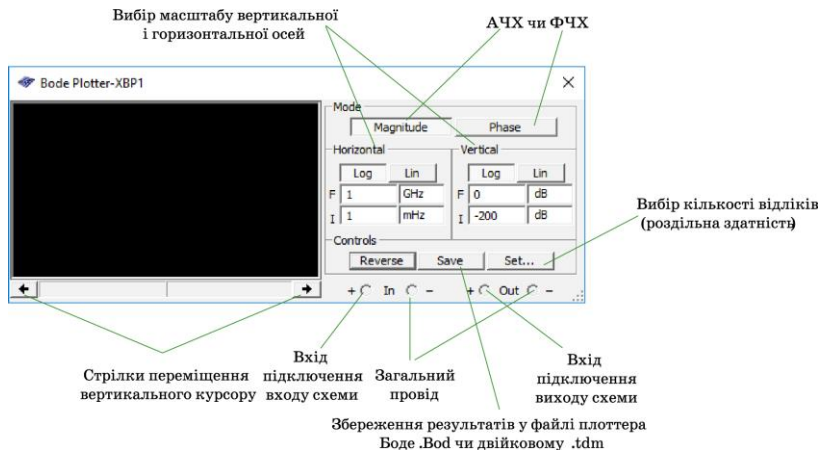


Рис. 2.33. Вікно відображення результатів та настроювання параметрів плотера Боде

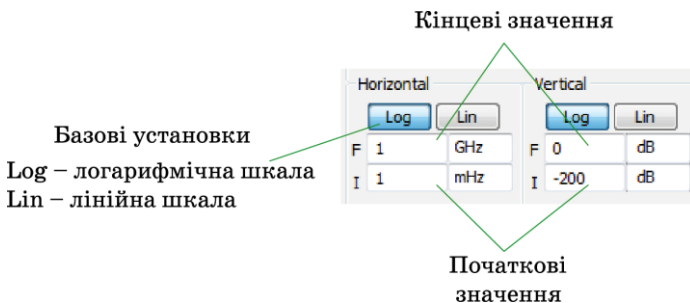


Рис. 2.34. Установки масштабу вертикальної та горизонтальної осей

В табл. 2.1 наведені діапазони вимірювань в лінійному та логарифмічному масштабах.

Перегляд результатів вимірювання плотером Боде.

Для ілюстрації вимірювання за допомогою Боде-плотера скористаємось моделлю 2_izm_magnitude.ms10, що наведена на рис. 2.35. Схема складається з коливального контуру з елементів L1 та C1 і резистора навантаження Rn. У вікні відображення ре-

зультатів спостерігається резонансна крива цього контуру. Для більш детального дослідження АЧХ діапазон аналізу встановлений від 10 кГц до 1 МГц, а мінімальне значення коефіцієнта передачі встановлено на рівні -50 дБ. Під час настоювання цих значень АЧХ на екрані перебудовується одночасно зі зміною параметра, що дозволяє вибрати найкращий масштаб для перегляду.

Табл. 2.1. Залежність одиниць і масштабу по вертикальній осі

Режим вимірювання	Використаний масштаб	Значення за замовчуванням	Максимальне кінцеве значення
Magnitude (gain)	Logarithmic	-200 дБ	200 дБ
Magnitude (gain)	Linear	0	10^9
Phase	Linear	-720°	720°

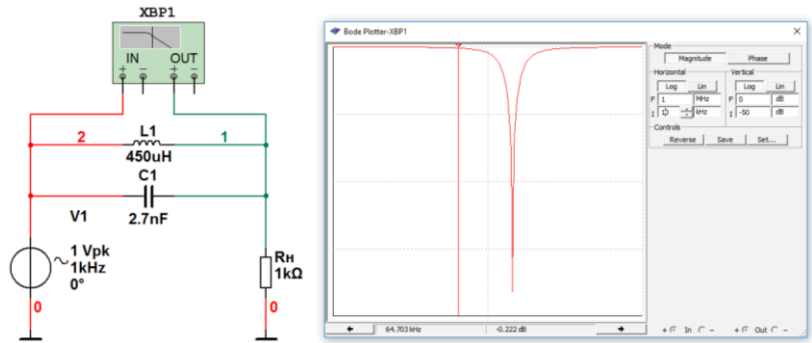


Рис. 2.35. Значення частоти і коефіцієнта передачі кола на перетині вертикального курсора і графіка

Для того, щоб прочитати значення частоти і коефіцієнта передачі схеми (або фазового зсуву) у будь-якій точці на графіку, потрібно змістити курсор, який знаходиться з лівого боку вікна відображення результатів вимірювання плотера Бode, у потрібну точку.

Для того, щоб перемістити вертикальний курсор, необхідно натиснути на стрілку в нижній частині плотера або пере-

тягнути вертикальний курсор від лівого краю дисплея плотера до точки на графіку, в якій необхідно виміряти значення відповідного параметра (рис. 2.35).

Частотна характеристика, що відображається на рис. 2.35, має дуже гострий резонанс, тому потрапити курсором в точку мінімального коефіцієнта передачі досить важко. Для точного переміщення до точки мінімуму або максимуму АЧХ потрібно розмістити курсор "м«ші" »а лінії курсору і натиснути праву кнопку "м«ші"».В результаті цього на екран буде виведено контекстне меню для переміщення курсору, в якому передбачено (рис. 2.36):

- переміщення курсору ліворуч або праворуч по осі X на задане значення частоти;
- переміщення до точки (ліворуч або праворуч) із заданим значення коефіцієнта передачі (або фазового зсуву при вимірюванні ФЧХ);
- переміщення до точки наступного максимуму (мінімуму) АЧХ ліворуч або праворуч.

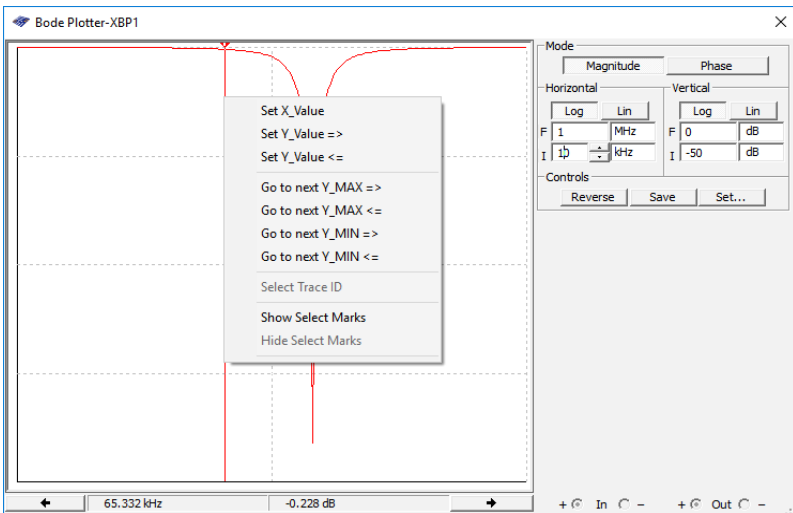


Рис. 2.36. Контекстне меню для переміщення курсора Бодє-плотера

Для точного вимірювання частоти резонансу можна скористатися пунктом контекстного меню Go to next Y_MIN зі

стрілкою, що вказує праворуч, і курсор автоматично переміститься до точки мінімального коефіцієнта передачі (рис. 2.37).

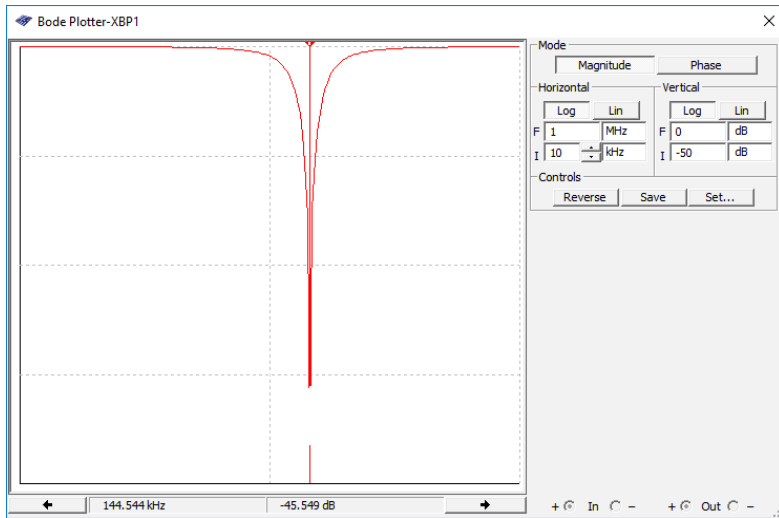


Рис. 2.37. Результати точного вимірювання частоти резонансу за допомогою контекстного меню

2.5.3. Генератор слова

Для формування тестових багаторозрядних цифрових сигналів використовується Word Generator (генератор слова), умовне позначення і вікно налаштувань параметрів якого показані на рис. 2.38. Генератор дозволяє формувати 32-розрядні кодові слова, які можна ввести з клавіатури безпосередньо у вікні введення коду.

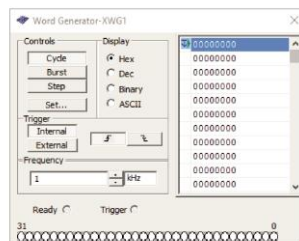
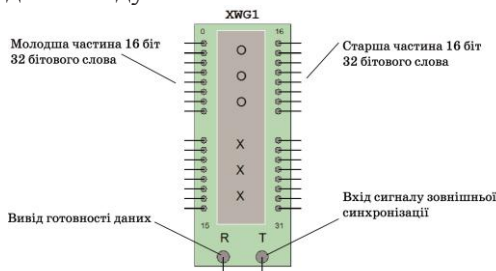


Рис. 2.38. Умовне позначення і вікно налаштувань параметрів генератора слова

Для вводу і збереження введених слів іншими способами необхідно натиснути кнопку Set у вікні настроювання параметрів генератора слова (рис. 2.39). У вікні попередніх настроювань у стовпці Preset Patterns можна вибрати один з восьми варіантів формування та збереження слова:

- No Change – не змінюється автоматично (ручне введення значень);
- Load – завантажити з файлу;
- Save – зберегти файл;
- Clear buffer – очистити буфер;
- Up Counter – автоматично збільшувати кожне наступне значення на 1;
- Down Counter – автоматично зменшувати кожне наступне значення на 1;
- Shift Right – зміщувати праворуч на одну позицію кожне наступне значення;
- Shift Left – зміщувати ліворуч на одну позицію кожне наступне значення.

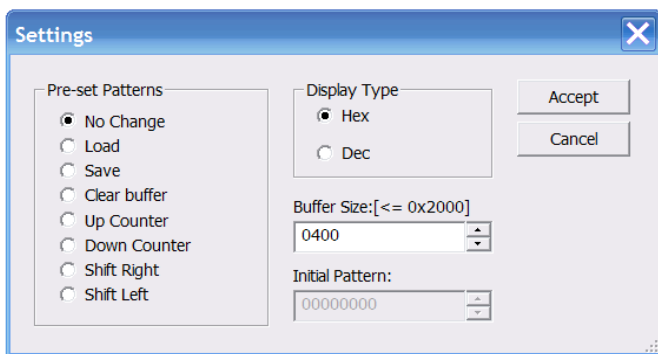


Рис. 2.39. Вікно настроювань генератора слова

Перед введенням кодів слів необхідно у вікні Buffer Size (рис. 2.39) ввести необхідну кількість кодів слів, яка задається в шістнадцятковому або десятковому коді залежно від типу виводу на дисплей – в шістнадцятковому (Hex) або десятковому (Dec) форматах, що задається у вікні Display Type. Гра-

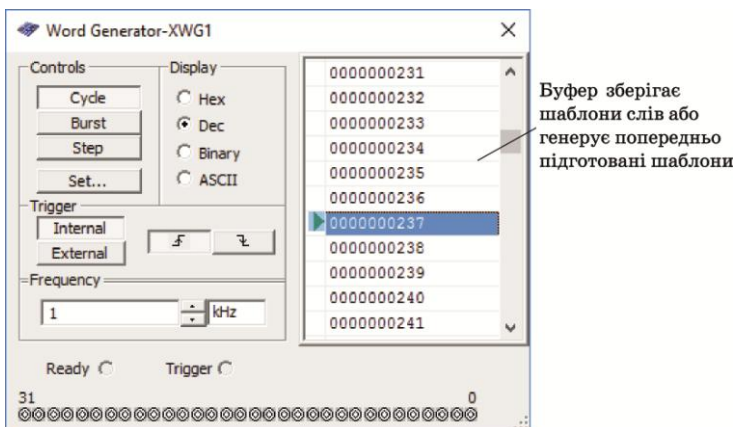
ничний розмір буфера складає 8192 кодових слова (2000 в шістнадцятковому коді).

При використанні попередніх настроювань генератора слова Up Counter, Down Counter, Shift Right та Shift Left у вікно Initial Pattern необхідно ввести початкове значення коду.

Розглянемо на прикладі, як буде виглядати набір кодів слів при таких налаштуваннях:

- Display Type – Dec;
- Buffer Size – 10;
- Shift Right;
- Initial Pattern – 2147483648 (відповідає шістнадцятковому коду 80000000).

Сформовані кодові слова показані на рис. 2.40. Для наочного представлення сформованого цифрового сигналу результат у вікні Word Generator представлений в десятковому коді.



Вихідні виводяться і відповідають тим, що відображаються на іконці генератора

Рис. 2.40. Основне вікно генератора слова

Щоб встановити об'єкт, значення і порядок зміни коду, необхідно натиснути кнопку Set (рис. 2.40) і у вікні Settings, що відкривається (рис. 2.41), провести необхідні установки.

До введення значень коду в будь-якому режимі необхідно вибрати, в якому коді відображати дані – шістнадцятковому (Hex) чи десятковому (Dec). Для цього необхідно задати об'єкт

буфера (не більше ніж 8192 символи) та номер початкової позиції, з якої почнеться зчитування коду (Initial Pattern), і натиснути кнопку Accept.

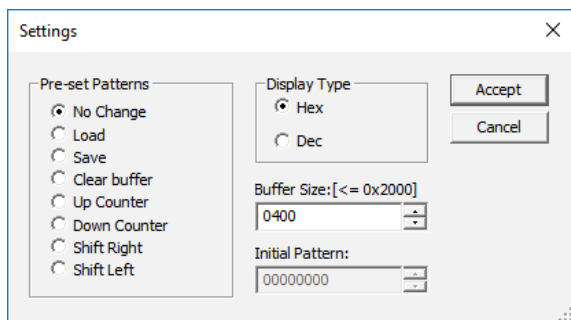


Рис. 2.41. Вікно Settings генератора слова

Для введення значень коду в режимі No Change курсором "м«ші" »необхідно клацнути по одній з кнопок вибору режиму формування коду (рис. 2.40): циклічному (Cycle), одноразовому (Burst) або покроковому (Step). В покроковому режимі після кожного натискання кнопки Run (на панелі інструментів кнопка позначена зеленим трикутником) виводиться чергове кодове слово. Після перебору всіх введених значень коду буде знову виводитися перше слово, потім друге і т. ін.

Після цього необхідно вибрати режим відображення в секції Display (Hex, Dec, Binary або ASCII) і ввести з клавіатури потрібні значення коду в кожний рядок. Подальші переміщення по полю екрану зручніше проводити не за допомогою "м«ші", а клавішами управління курсором. Вміст екрану можна стерти і завантажити нове значення (або записати результати у файл). Для запису необхідно натиснути кнопки Save і Accept, попередньо натиснувши кнопку Set (рис. 2.41). В діалоговому вікні необхідно вказати ім'я файлу (за замовчуванням пропонується ім'я схемного файлу). В отриманому таким чином текстовому файлі з розширенням .dp будуть записані у вигляді таблиці всі набрані на екрані комбінації. За необхідністю файл можна відредагувати в будь-якому текстовому редакторі і завантажити знову натисканням кнопки Load.

Окрім режиму ручного вводу коду передбачена можли-

вість автоматичного формування коду відповідно до змін коду на виході підсумовувального (Up Counter) або віднімального (Down Counter) двійкового лічильника, а також автоматичного зміщення коду на одну позицію двійкового коду (тобто зміна у 2 рази) ліворуч (Shift Left) або праворуч (Shift Right).

Для введення і збереження введених слів іншими способами необхідно натиснути кнопку Set у вікні настроювання параметрів генератора слова (рис. 2.41). При цьому у вікні попередніх настроювань у стовпці Preset Patterns необхідно вибрати один з восьми варіантів формування та збереження слова (розглянуто раніше).

На вихід R (Ready) генератора слова виводиться сигнал тактової частоти, який може бути використаний для синхронізації інших пристроїв. Сигнал з цього виходу супроводжує кожну кодову комбінацію, що подається на вихід і може бути використаний в тому випадку, коли досліджуваний пристрій має властивість квітування (підтвердження). В цьому випадку після отримання чергової кодової комбінації і супроводжуючого його сигналу READY досліджуваний пристрій формує сигнал підтвердження отримання даних, який подається на вхід зовнішньої синхронізації генератора T і виробляє черговий запуск генератора, якщо він працює в режимі зовнішньої синхронізації.

На рис. 2.42 наведено приклад формування 4-розрядного цифрового сигналу з періодом повторення 16 тактів (режим виводу Cycle) і часові діаграми цього сигналу, отримані за допомогою чотириканального осцилографа.

Текст файлу, що містить інформацію про сформовані сигнали, наведено на рис. 2.43. Частота виводу сигналів задається в рядку Frequency (рис. 2.40). За замовчуванням задається частота виведення сигналу 1 кГц.

Сформовані слова знімаються з чотирьох виходів (від 0 до 3) молодших розрядів XWG1 (рис. 2.42). На інших виходах формуються нулі відповідно до таблиці програмування генератора. Передбачена можливість підключення зовнішнього сигналу синхронізації по входу T. За замовчуванням включена синхронізація від внутрішнього тактового генератора.

Для очищення буфера від введених кодових слів необ-

хідно натиснути кнопку Clear buffer.

Генератор кодових слів зручно використовувати для тестування різних цифрових пристроїв, тому що він дозволяє формувати довільні послідовності нулів та одиниць.

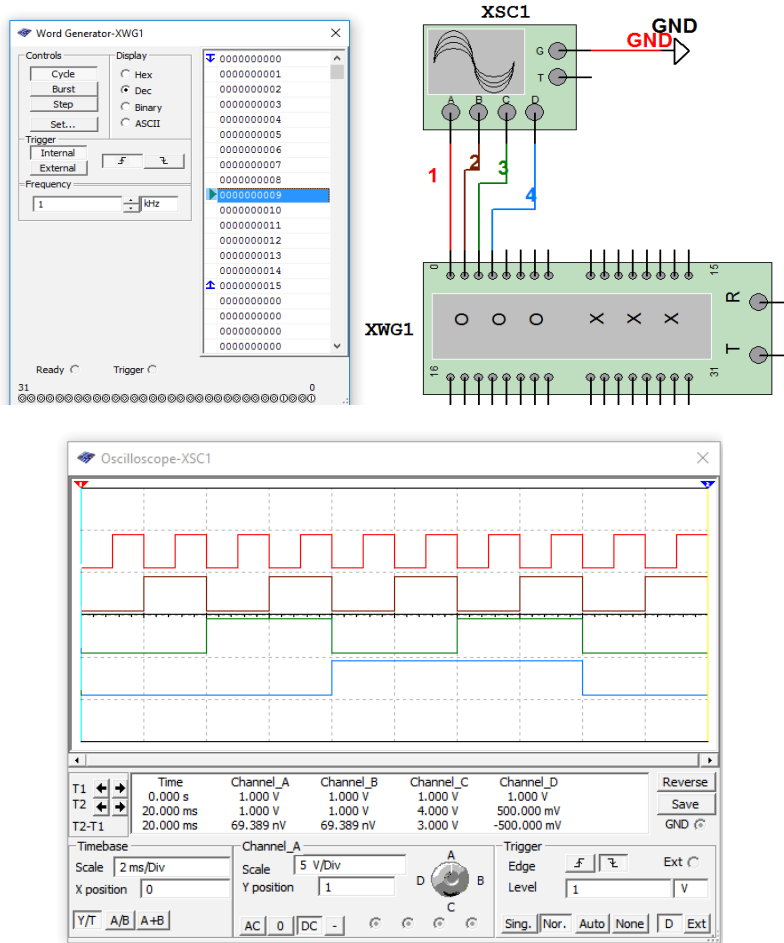


Рис. 2.42. Формування 4-розрядного двійкового зростаючого коду генератором слова в циклічному режимі

2.5.4. Аналізатор спектру

Аналізатор спектру призначений для вимірювання спектрів сигналів. Аналіз спектру відбувається з використання швидкого перетворення Фур'є' (ШПФ). На рис. 2.44 наведений зовнішній вигляд та органи керування аналізатором спектру.

Перед проведенням аналізу необхідно задати параметри спектрального аналізу:

- діапазон частот аналізу (Span);
- початкове і кінцеве значення частоти (Start та End);
- центральну частоту (Center);
- динамічний діапазон аналізу (Range), який задається як ціна поділки по вертикалі у В/поділку або дБ/поділку;
- опорний рівень в дБ і розділову здатність аналізу (Resolution Freq.);
- кількість точок спектрального аналізу, максимальний рівень вхідного сигналу і джерело синхросигналу (внутрішнє або зовнішнє) задають, натиснувши кнопку Set.

Data:
00000000
00000001
00000002
00000003
00000004
00000005
00000006
00000007
00000008
00000009
0000000A
0000000B
0000000C
0000000D
0000000E
0000000F
Initial:
0000
Final:
000F

Рис. 2.43. Таблиця, що зберігається в файлі word.dp

Чим менше значення розділової здатності, тим точніше значення спектру. Розділова здатність аналізу також залежить від кількості усереднених реалізацій сигналу, яка задається після натискання кнопки Set у вікні (рис. 2.44).

Чим більша кількість усереднень, тим більша точність аналізу, однак і більший час для його проведення. Значення кількості усереднень може вибиратись в діапазоні від 1024 до 32768. Після вибору потрібної кількості усереднень потрібно натиснути кнопку Accept. Після цього можна проводити аналіз, натиснувши кнопку початку моделювання у Multisim.

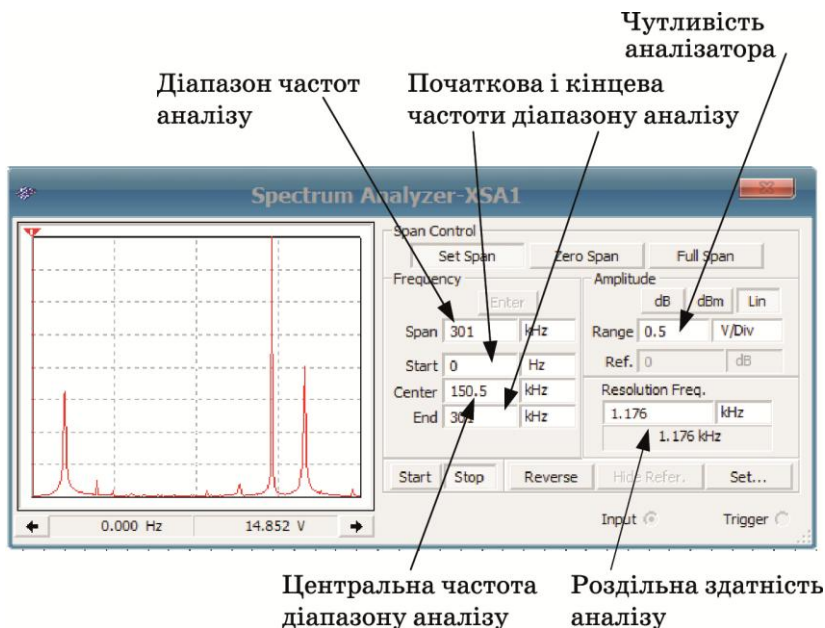


Рис. 2.44. Органи керування аналізатором спектру

Ілюстрацію роботи аналізатора спектру покажемо на прикладі аналізу спектра послідовності однополярних прямокутних імпульсів. Модель для дослідження спектру 2_spectr_imp.ms10 наведена на рис. 2.45 і складається всього з двох елементів – генератора прямокутних однополярних імпульсів V1 (CLOCK VOLTAGE) і резистора R1.

Для аналізу подаємо на вхід аналізатора спектру послідовність прямокутних імпульсів з амплітудою 5 В і частотою 1 кГц, тривалість яких складає 50% від тривалості періоду.

Аналіз отриманої спектрограми (рис. 2.45) показує, що в спектрі в заданому діапазоні аналізу від 0 Гц до 30 кГц присутній цілий ряд непарних гармонік. Парні гармоніки у спектрі відсутні, що видно в точці розташування курсору.

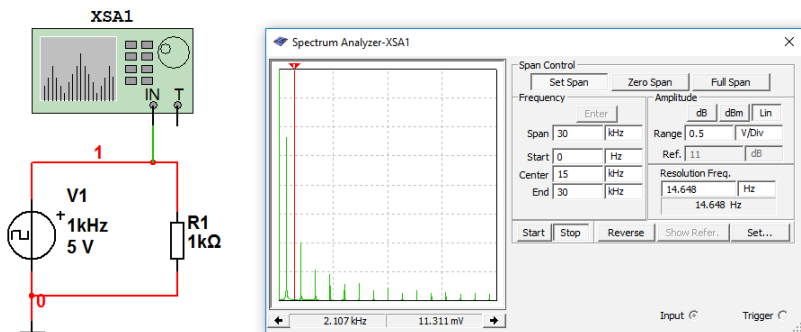


Рис. 2.45. Схема моделі та результати аналізу спектру послідовності прямокутних імпульсів з частотою слідування 1 кГц та коефіцієнтом заповнення 50%

2.5.5. Логічний аналізатор

Логічний аналізатор призначений для одночасного відображення на одному екрані до 16 процесів, що протікають в цифрових схемах. Щоб використати інструмент, потрібно натиснути на кнопку Logic Analyzer на панелі інструментів і перемістити іконку на робочій лист. Іконка використовується для підключення логічного аналізатора до схеми. Подвійне клацання по іконці відкриває панель інструмента, яка використовується для введення установок і перегляду результатів вимірювання.

Логічний аналізатор та призначення його входів наведені на рис. 2.46, а вікно настроювання та відображення результатів аналізу – на рис. 2.47.

У цьому вікні 16 кіл з лівого боку показують під'єднані входи. Коли входи з'єднані з вузлами схеми, кола відображаються з чорними крапками всередині, а поруч з ними відображаються імена вузлів, при цьому колір діаграми на даному вході збігається з кольором проводу на схемі, який під'єднаний до цього входу. Вільні входи кіл виводів відображаються без чорних крапок, а діаграми сигналів біля цих входів не відображаються (рис. 2.47).

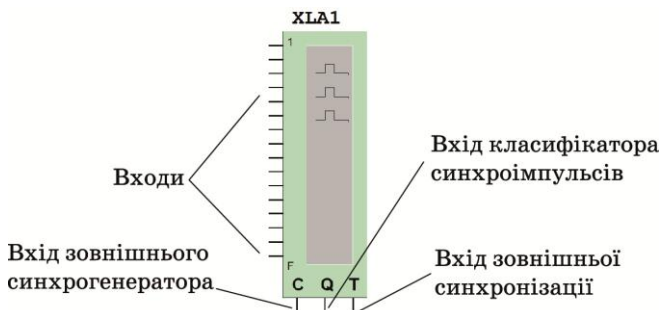


Рис. 2.46. Зовнішній вигляд іконки логічного аналізатора

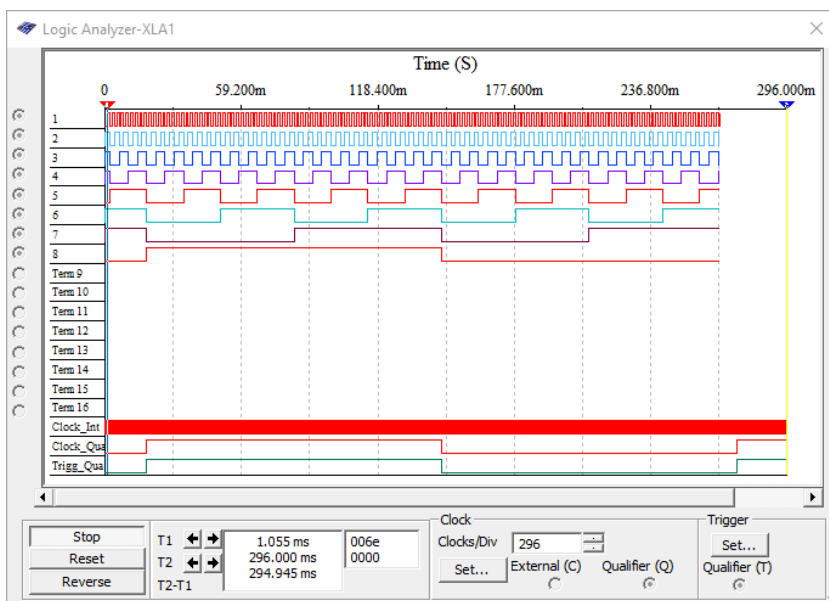


Рис. 2.47. Вікно налаштування та відображення результатів аналізу логічного аналізатора

В центральній частині панелі керування знаходиться вікно показань курсорів, в якому розташовано три поля:

- "T«" »показання курсора T1);
- "T«" »показання курсора T2);
- "T«-T1" »часовий зсув між курсорами).

Кнопки стрілок дозволяють змінювати положення кур-

сорів. Часове положення сигналів можна виміряти в точках перетину курсорів T1 і T2, коли робота аналізатора зупинена. Також автоматично перший курсор T1 переміщується в точку моменту синхронізації, коли робота аналізатора зупиняється.

Коли схема активується, логічний аналізатор записує сигнали, що подаються на його входи, і відображає їх у вигляді часових діаграм. Сигнали відображаються як прямокутні імпульси. Верхня діаграма відображає сигнал каналу 1, наступна – каналу 2 та ін. У цьому ж вікні також відображається сигнал внутрішнього або зовнішнього тактового генератора, сигнал зовнішньої синхронізації і сигнал на вході класифікатора синхроімпульсів (рис. 2.46).

Логічний аналізатор збирає і виводить дані, доки не досягне кількості відліків, яка задається користувачем після натискання кнопки Set.

Після вмикання режиму симуляції логічний аналізатор автоматично починає проводити аналіз. Для перезапуску приладу потрібно натиснути кнопку Reset. Це призведе до очистки всієї інформації у вікні логічного аналізатора і початку нового циклу збору інформації.

Щоб відображати збережені дані без їх подальшої зміни, потрібно натиснути кнопку Stop. Щоб знову почати процес аналізу даних після зупинки аналізатора, потрібно натиснути кнопку Reset або знову запустити процес симуляції Multisim.

Для правильного відображення сигналів, що подаються на входи логічного аналізатора, потрібно враховувати принцип його роботи. Він полягає у тому, що з вхідного сигналу зчитуються відліки з частотою, яка задається користувачем. Сигнал цієї частоти у логічному аналізаторі має назву Clock rate (частота дискретизації) і задається від внутрішнього або зовнішнього генератора. Частота цього сигналу має перевищувати частоту найбільш високочастотного вхідного сигналу не менше ніж у 10 разів. Однак її потрібно вибирати набагато більшою для випадку, якщо потрібно досліджувати короткі імпульси. Приблизно частоту сигналу Clock rate можна розрахувати за такою формулою $f_{clk} \geq 1/t_{i,min}$, де $t_{i,min}$ – тривалість найкоротшого імпульсу.

Встановлення параметрів тактового генератора логі-

чного аналізатора.

Генератор сигналів Clock rate може бути внутрішнім або зовнішнім. Для налаштування установок тактового генератора потрібно виконати декілька простих операцій.

1. Натиснути кнопку Set в області Clock логічного аналізатора (рис. 2.47). При цьому з'явиться діалогове вікно Clock Setup (рис. 2.48).

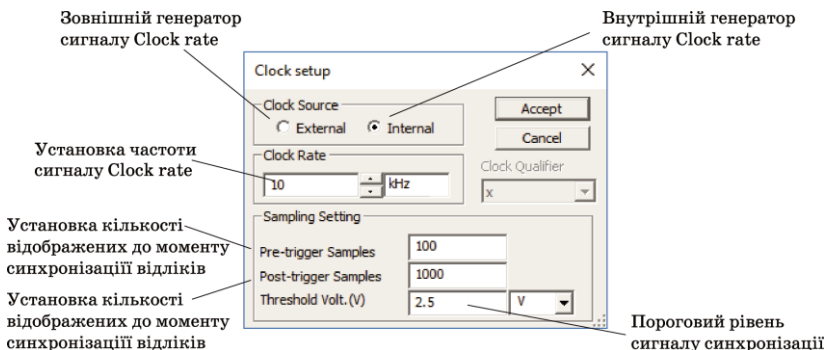


Рис. 2.48. Діалогове вікно Clock Setup

2. Вибрати режим роботи із зовнішнім або внутрішнім тактовим генератором.

3. Встановити частоту тактового генератора, яка має бути набагато більшою за частоту самого височастотного сигналу на входах аналізатора. Краще вибирати значення цієї частоти у 10...100 разів більшим за частоту сигналу.

Класифікатор сигналу зовнішньої синхронізації (Clock Qualifier) дозволяє змінювати рівень спрацьовування від сигналу синхронізації. Якщо він встановлений в "X« »рис. 2.48), класифікатор вимкнений і не впливає на роботу аналізатора. Якщо він встановлений в "1« »бо "0«,»то синхронізація буде відбуватися в той момент часу, коли рівень сигналу синхронізації збігається з заданим класифікатором сигналом.

4. Встановити кількість відліків, яку потрібно показувати до моменту синхронізації (Pre-trigger samples) і після нього (Post-trigger samples). Кількість відліків для відображення можна приблизно розрахувати, знаючи кількість періодів сигналу, які потрібно відображати у вікні аналізатора.

Розглянемо, як це зробити, на прикладі. Нехай потрібно відобразити дещо більше, ніж повний цикл зміни 8-розрядного коду, що відповідає 256 періодам сигналів. Додаємо ще 40 періодів і отримаємо загальну кількість 296, що вводимо у вікно Clocs/Div (рис. 2.47). Після цього натискаємо кнопку Set і у вікні Clock Setup (рис. 2.48) потрібно задати кількість Pre-trigger samples і Post-trigger samples. Для цього потрібно помножити кількість періодів сигналу на відношення частот сигналів Clock rate і найбільш високочастотного сигналу на вході аналізатора. Для розглянутого на рис. 2.47 прикладу це відношення складає 10 і тому кількість Post-trigger samples має бути встановлена рівною або більше, ніж 2560. Значення Pre-trigger samples потрібно встановити 40.

5. Для запам'ятовування встановлених параметрів клацнути по кнопці Асепт.

Встановлення параметрів синхронізації

Логічний аналізатор може бути настроєний або на перемикання по читанню заданого слова або комбінації слів, або фронту чи спаду сигналу синхронізації.

Для того, щоб задати до трьох слів, при яких відбувається синхронізація, або комбінації слів, необхідно виконати такі дії.

1. Клацнути по Set в області Trigger логічного аналізатора. Відкриється вікно Trigger Settings (рис. 2.49).

2. Вибрати Positive, Negative або Both для позитивного і негативного фронтів тактового сигналу.

3. Клацнути в області, яка позначена Pattern A, Pattern B або Pattern C, і ввести двійкове число. "X" означає або 1, або 0. Можливі комбінації сигналу синхронізації роботи аналізатора наведені на рис. 2.50.

4. З випадаючого списку Trigger Combination вибрати потрібну комбінацію.

5. З випадаючого списку Trigger Qualifier вибрати потрібний рівень сигналу синхронізації – 0 або 1.

6. Натиснути кнопку ОК.

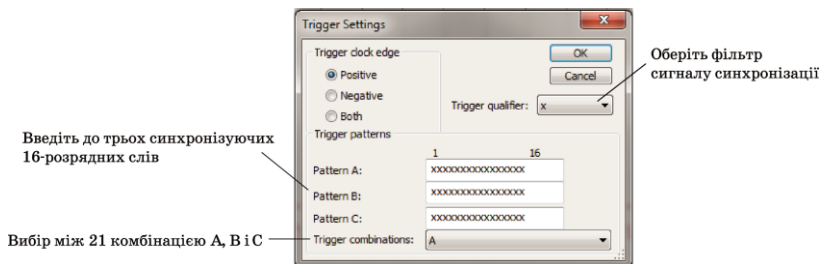


Рис. 2.49. Діалогове вікно *Trigger Settings*

A	B	C
A OR B	A OR C	B OR C
A OR B OR C	A AND B	A AND C
A AND C	A AND B AND C	A NOT B
A NOT C	B NOT C	A THEN B
A THEN C	B THEN C	(A OR B) THEN C
A THEN (B OR C)	A THEN B THEN C	A THEN (B WITHOUT C)

Рис. 2.50. Можливі комбінації сигналу синхронізації роботи аналізатора

Приклад використання логічного аналізатора.

Проілюструємо роботу з логічним аналізатором на прикладі. Для аналізу сформуємо восьмирозрядний зростаючий код за допомогою генератора слова (рис. 2.51).

Для моделювання була задана частота вхідного сигналу 1 кГц, а тактова частота внутрішнього генератора аналізатора – 10 кГц. Для розглянутого прикладу кожному періоду вхідного сигналу відповідає 10 відліків (samples). З урахуванням цього у вікно Post-trigger sample необхідно ввести число не менше 2560 (у прикладі введене число 2660) для того, щоб на екран був виведений один повний цикл роботи генератора слова. Більш зручно розглядати процеси, коли на екран виводиться не один період, щоб бачити початок і закінчення процесу, а дещо більше. Для цього і введено число 2660 відліків.

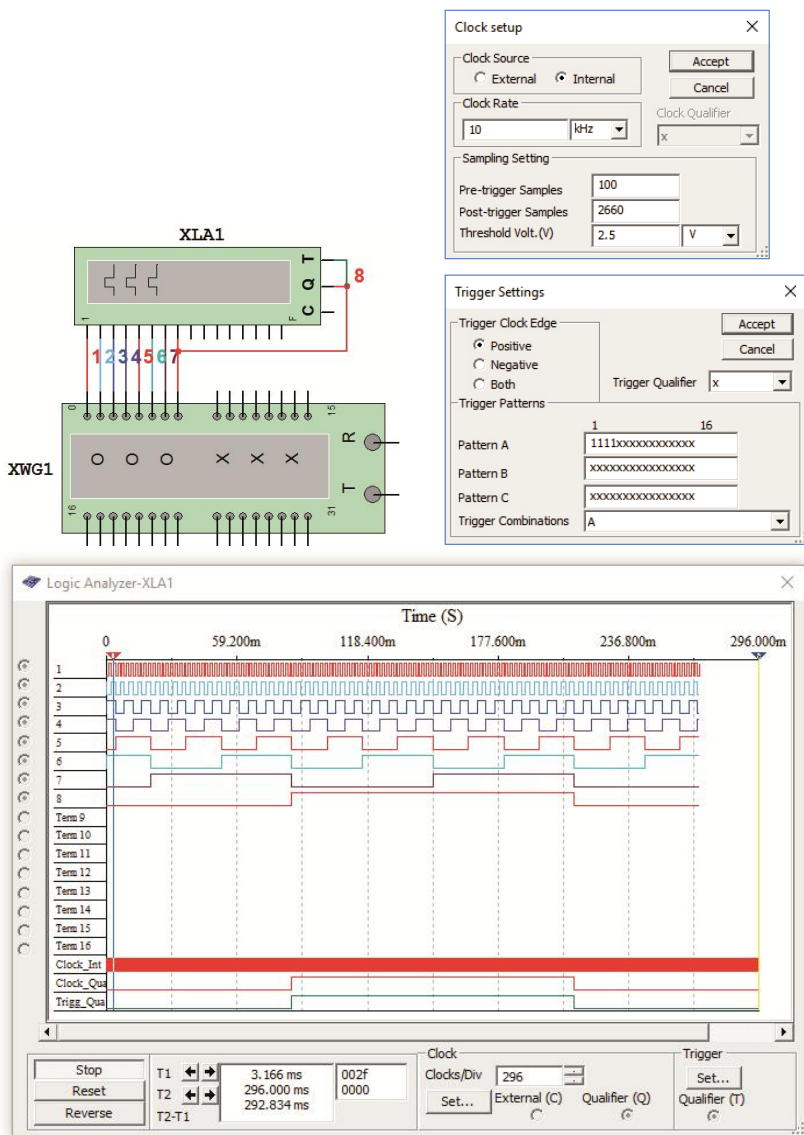


Рис. 2.51. Схема підключення, часові діаграми та установки параметрів логічного аналізатора

Масштаб по осі часу можна змінювати, задаючи число

тактів на ділення (Clocs/Div), яке задається у вікні виводу результатів аналізу (рис. 2.47). Чим менше задане число, тим більше масштаб по осі часу. У розглянутому прикладі масштаб встановлений на рівні 296 відліків на поділці.

Контрольні питання

1. Чи змінюється полярність постійних аналогових сигналів?
2. Які сигнали називають двійковими?
3. Чи потребують індивідуального регулювання і настроювання вузли цифрових пристроїв?
4. Наведіть відмінності між відео та радіоімпульсами.
5. Дайте пояснення параметрам імпульсу та імпульсної послідовності.
6. У якій послідовності прямокутних імпульсів буде ширшим спектр: послідовності імпульсів з $Q = 2$ або з $Q = 5$? Покажіть це, використавши модель 2_spectr_imp.ms10.
7. За допомогою якого виду аналізу можна дослідити спектр сигналу в програмі NI Multisim?
8. Крім графіку спектру які ще дані дозволяє отримати аналіз Фур'є?
9. Яким приладом можна дослідити спектр сигналу в програмі NI Multisim?
10. Яким приладом можна дослідити часові діаграми цифрових сигналів, якщо їх більше 4-х? Якщо менше 4-х?
11. Якщо досліджуються два сигнали різної частоти за допомогою осцилографа, яким сигналом треба здійснювати синхронізацію осцилографа – з більш високою частотою, чи з більш низькою? Проілюструйте це, використовуючи модель 2_word generator.ms10.

3. ПРОХОДЖЕННЯ ІМПУЛЬСНИХ СИГНАЛІВ ЧЕРЕЗ RC – КОЛА

3.1. Диференціювальні, інтегрувальні та розділові RC-кола

Коло, утворене лінійними резистором і конденсатором, називається лінійним RC-колом. Залежно від способу включення елементів R і C розрізняють два кола – інтегрувальне та диференціювальне (рис. 3.1).

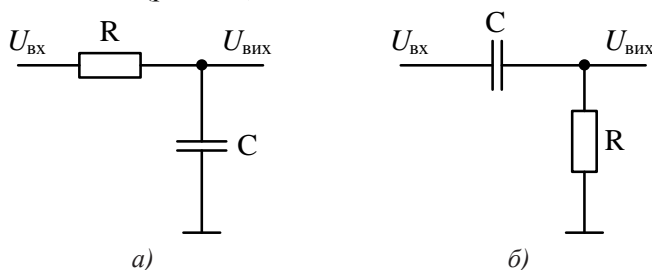


Рис. 3.1. Інтегрувальне (а) та диференціювальне (б) коло

Розглянемо реакцію RC-кола при дії імпульсів прямокутної форми. Нехай на вхід RC-кола подається поодинокий імпульс прямокутної форми (рис. 3.2). Для знаходження форм напруг $U_C(t)$ та $U_R(t)$ на виході кола використовується метод, який полягає в тому, що вхідний прямокутний імпульс розкладають на дві елементарні напруги $U_{вх1}(t)$ та $U_{вх2}(t)$, що стрибкоподібно змінюються, визначають форму напруг на виході при дії цих елементарних напруг, а далі методом накладання знаходять напруги на резисторі та конденсаторі.

На рис. 3.3 наведені побудовані вказаним методом часові діаграми напруг $U_C(t)$ та $U_R(t)$ при різних співвідношеннях між сталою часу кола τ і тривалістю вхідного імпульсу t_i . З рис. 3.3, а виходить, що при малому у порівнянні з t_i значенні τ форма напруги на ємності $U_C(t)$ виявляється близькою до форми вхідних імпульсів $U_{вх}(t)$.

При $\tau/t_i < 0.03$ тривалість фронту напруги $U_C(t)$, яка дорів-

ное 3τ , виявляється меншою $t_i/10$, і форма вихідного імпульсу може вважатися майже прямокутною. При збільшенні τ/t_i тривалість фронту $U_C(t)$ росте і при $\tau/t_i > 0.3$ напруга $U_C(t)$ не встигає за час t_i збільшитися до стаціонарного значення. Форма $U_C(t)$ виявляється при цьому близькою до пілоподібної. При подальшому збільшенні τ/t_i амплітуда напруги $U_C(t)$ зменшується.

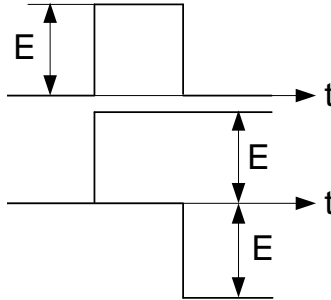


Рис. 3.2. Розкладання прямокутного імпульсу на дві стрибкоподібні дії

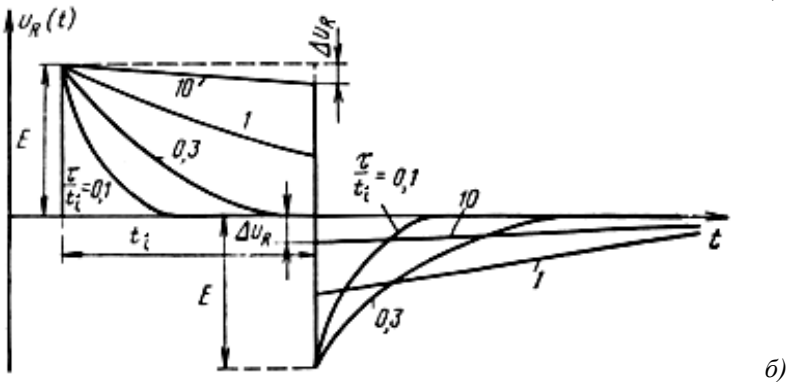
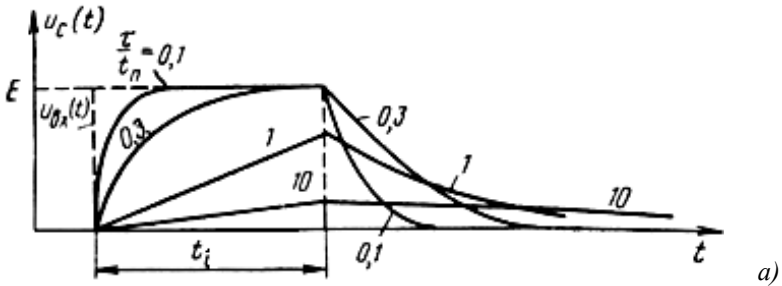


Рис. 3.3. Часові діаграми напруг на різних елементах RC-кола

З рис. 3.3, б слідує, що при великих значеннях τ/t_i форма напруги $U_R(t)$ на резисторі виявляється близькою до форми $U_{\text{вх}}(t)$. При цьому спостерігаються спад вершини імпульсу $U_R(t)$ і негативний викид після його закінчення. Величини спаду і негативного викиду зменшуються при збільшенні відношення τ/t_i . При малих значеннях відношення τ/t_i форма напруги $U_R(t)$ представляє собою два імпульси загостреної форми, початки яких збігаються за часом з перепадами вхідної напруги і мають полярність цих перепадів. Амплітуда імпульсів виявляється рівною амплітуді E вхідної напруги, а тривалість – більше 3τ .

Для випадку періодичних імпульсів з порівняно великою шпаруватістю, коли перехідні процеси в RC-колі, викликані дією попереднього імпульсу, встигають практично закінчитися до моменту приходу наступного імпульсу, можна використовувати підхід, описаний вище. У випадку, якщо стала часу кола τ сумірна або перевищує паузу між імпульсами, картина процесів відрізняється від розглянутої вище.

Припустимо, що в момент $t=0$ до RC-кола підключається джерело напруги прямокутної форми (рис. 3.4, а). Нехай напруга $U_C(t)=0$ при $t<0$, а величина τ значно перевищує період T повторення імпульсів. Під час першого імпульсу ємність C заряджається до деякої напруги. У паузі між першим і другим імпульсами ємність розряджається, проте на початок другого імпульсу вона не встигає розрядитися повністю і на ній залишається деяка напруга $U_C(t)$. Під час другого імпульсу ємність знову заряджається, але до більшого значення, ніж під час першого імпульсу, а в паузі знов розряджається, однак не повністю. Приріст напруги на ємності ΔU_k за час k -го імпульсу дорівнює

$$\Delta U_k = (E - U_{Ck-1}) \left(1 - e^{-\frac{t_i}{\tau}}\right),$$

де U_{Ck-1} – напруга на ємності після закінчення $(k-1)$ -го періоду.

Якщо $\tau/t_i \ll 1$, то формула може бути переписана у такому наближеному вигляді:

$$\Delta U_k = (E - U_{Ck-1}) \frac{t_i}{\tau}.$$

За час паузи між k -им та $k+1$ -им імпульсами ємність розряджається на величину напруги

$$\Delta U_{pk} = (U_{Ck-1} + \Delta U_k) \left(1 - e^{-\frac{T-t_i}{\tau}}\right) \approx (U_{Ck-1} + \Delta U_k) \frac{T-t_i}{\tau}.$$

На початку процесу після включення генератора вхідної напруги величина напруги на ємності U_{Ck-1} мала і приріст напруги ΔU_k перевищує спад ΔU_{pk} . Тому від періоду до періоду напруга на конденсаторі збільшується (рис. 3.4, б).

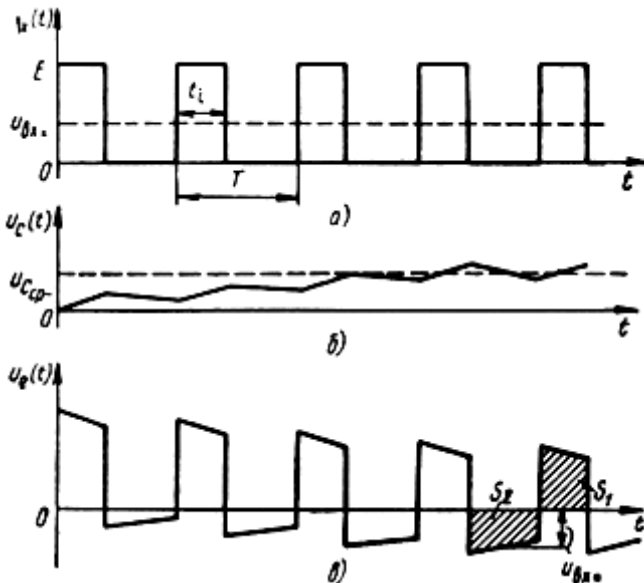


Рис. 3.4. Часові діаграми роботи розділового кола

Проте з часом при збільшенні напруги на конденсаторі U_{Ck-1} різниця напруг $E - U_{Ck-1}$ і величина ΔU_k зменшуються, а значення ΔU_{pk} росте. Внаслідок цього після закінчення певного часу в колі встановлюється динамічна рівновага, при якій приріст напруги ΔU_k під час заряду дорівнює спаду ΔU_{pk} під час розряду. Середнє значення напруги на конденсаторі в усталеному режимі виявляється рівним постійній складовій вхідної напруги $U_{ox}(t)$.

Такі кола називаються розділовими і призначені для розділення постійної та змінної складових. Форма сигналу на вихо-

ді такого кола повторює форму вхідного сигналу, але постійна складова вихідного сигналу дорівнює нулю.

Вхідні сигнали можуть бути двох видів:

- без постійної складової (рис. 3.5, а);
- з постійною складовою (рис. 3.5, б).

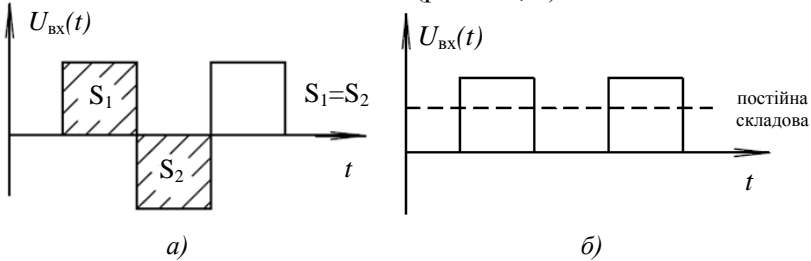


Рис. 3.5. Сигнали без постійної (а) та з постійною складовою (б)

Постійна складова – це площа над прямою, що описує сигнал.

Головною умовою при розділенні сигналу є те, що площі, обмежені позитивними і негативними імпульсами, мають бути однаковими.

Напруги та струм у RC-колі (рис. 3.6) під впливом одиничного стрибка (рис. 3.7) залежать від сталої часу кола τ .

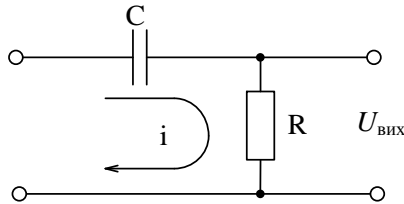


Рис. 3.6. Електрична схема RC-кола

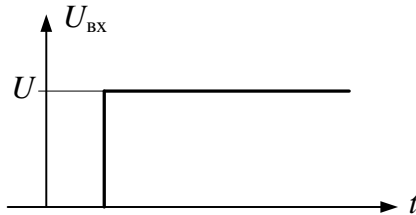


Рис. 3.7. Одиничний стрибок напруги

Напруга на конденсаторі C визначається таким чином:

$$U_c = U \cdot (1 - e^{-\frac{t}{\tau}}). \quad (3.1)$$

Залежності $U_c(t)$ при різних значеннях сталої часу RC -кола наведені на рис.3.8.

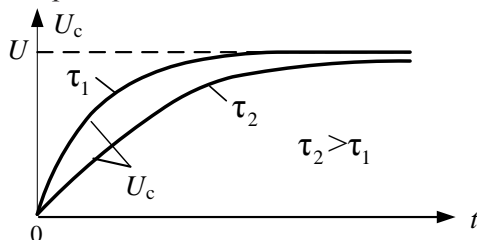


Рис. 3.8. Напруга на конденсаторі при різних значеннях сталої часу

Напруга на виході RC -кола (на резисторі) має вигляд:

$$U_R = U - U_c = U - U + U \cdot e^{-\frac{t}{\tau}} = U \cdot e^{-\frac{t}{\tau}}. \quad (3.2)$$

Залежності $U_R(t)$ при різних значеннях τ наведені на рис. 3.9. Оскільки $U_R = i \cdot R$, то:

$$i = \frac{U_R}{R} = \frac{U}{R} \cdot e^{-\frac{t}{\tau}} \quad (3.3)$$

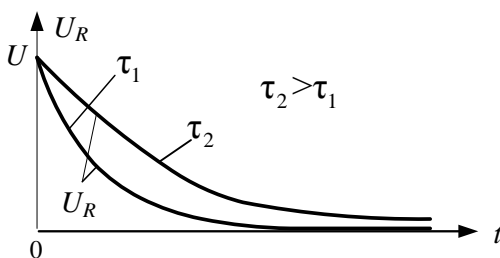


Рис. 3.9. Напруга на резисторі при різних значеннях сталої часу τ

Залежність $I(t)$ наведена на рис. 3.10.

Диференціальні та розділові RC -кола.

Диференціальним колом називають таке коло, сигнал на виході якого має значення, пропорційні в кожен момент часу

похідний від вхідного сигналу.

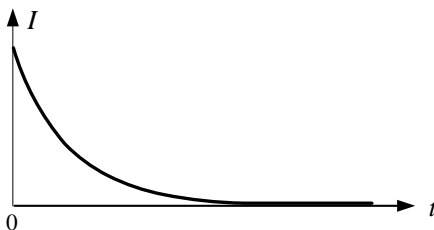


Рис. 3.10. Залежність струму через резистор від часу

Отже, $u_{\text{вих}}(t) = K \cdot \frac{du_{\text{вх}}(t)}{dt}$. Коефіцієнт K має розмір-

ність часу (с), в іншому випадку розмірність лівої і правої частин рівняння не буде однаковою. Ідеальним пристроєм диференціювання можна вважати конденсатор C або котушку L . Наприклад, при використанні конденсатора C можна вважати вхідним сигналом напругу на ньому $u_{\text{вх}}(t)$, а вихідним – струм $i(t)$

в колі. Ці змінні зв'язані відомим виразом $i(t) = C \cdot \frac{du_{\text{вх}}(t)}{dt}$, тоб-

то струм в колі прямо пропорційний похідній від вхідної напруги. Однак використовувати таку схему на практиці не можна, тому що вона не містить елемента, який забезпечував би обмеження струму, тобто реєстрацію його значень.

Для того, щоб отримати вихідний сигнал у формі, зручній для спостереження або реєстрації, у коло послідовно включають чутливий до струму прилад з внутрішнім опором R . У найпростішому випадку це може бути резистор R , напруга на якому пропорційна струму $U_R = i \cdot R$. Розглянуте RC -коло може виконувати функції диференціювання (вкорочення) при $\tau \ll t_I$, або бути розділовим, якщо $\tau \gg t_I$ (t_I – тривалість імпульсу).

На рис. 3.11 показані графіки напруг U_C і U_R такого кола в режимі диференціювання ($\tau \ll t_I$) та в режимі розділення ($\tau \gg t_I$).

Розглянемо диференціальне коло під впливом імпуль-

сної послідовності (рис.3.12).

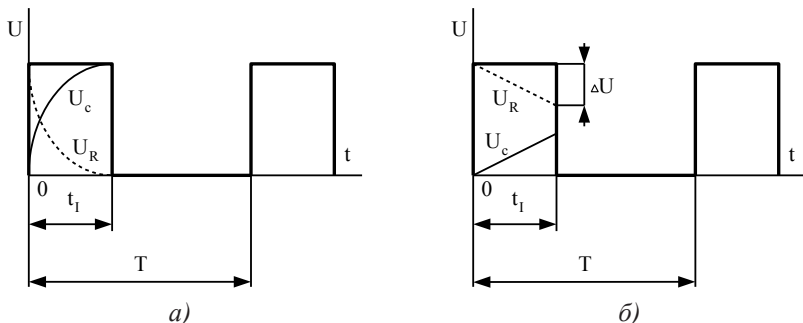


Рис. 3.11. Графіки напруги при $\tau \ll t_I$ (а) та $\tau \gg t_I$ (б)

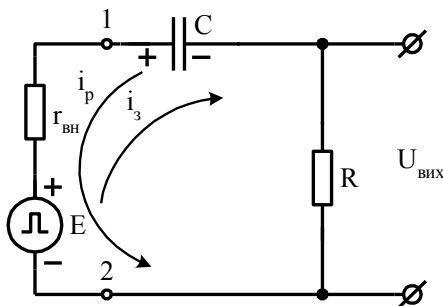


Рис. 3.12. Електрична схема диференціовального RC-кола

Якщо на вхід кола подати імпульс, конденсатор C заряджається зарядним струмом i_3 , а під час паузи між імпульсами – розряджається, зумовлюючи розрядний струм i_p (при цьому $E = 0$).

Припустимо, що $r_{вн} \ll R$, тоді їм можна знехтувати ($r_{вн} = 0$). Розглянемо режим I при $\tau \ll t_I$, $\tau \ll t_n$. Після закінчення імпульсу (момент часу t_1) $E = 0 \Rightarrow U_R = -U_c$ (рис. 3.13).

Під час паузи ($t_1 - t_2$) розряд конденсатора C відбувається повністю (рис.3.13), тому що $\tau \ll t_n$;

$$U_{вих} = i_p \cdot R = R \cdot C \cdot \frac{dU_c}{dt}; \quad U_c = U_{12} - U_{вих}.$$

Тоді

$$U_{\text{вих}} = R \cdot C \cdot \frac{d(U_{12} - U_{\text{вих}})}{dt}.$$

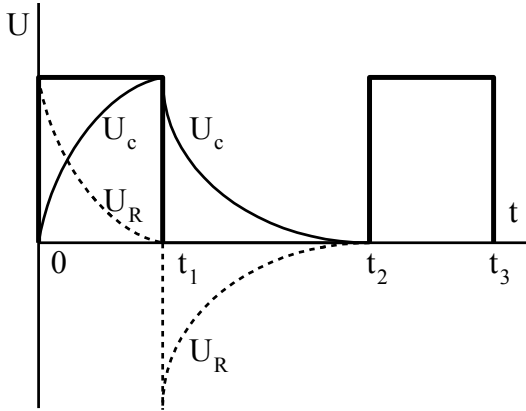


Рис. 3.13. Залежності $U_C(t)$ і $U_R(t)$; режим I при $\tau \ll t_I$, $\tau \ll t_n$

При $U_{\text{вих}} \ll U_{12} = U_{\text{вх}}$ отримаємо:

$$U_{\text{вих}} = R \cdot C \cdot \frac{dU_{\text{вх}}}{dt}.$$

Таким чином, отримали ідеальне диференціальне коло. Отже, для того, щоб коло було диференціальним, необхідно виконання трьох умов:

$$\tau \ll t_I; \quad \tau \ll t_n; \quad U_{\text{вих}} \ll U_{12} = U_{\text{вх}}.$$

При цьому графік напруги $U_{\text{вих}}$ при наявності імпульсної послідовності на вході буде мати такий вигляд (рис. 3.14).

Режим, при якому $\tau \ll t_I$ та $\tau \gg t_n$, не використовується для диференціювання, тому що конденсатор C не встигає розрядитися до нуля за час t_n , тому нульові початкові умови не виконуються.

Аналогічно в режимі II при $\tau \gg t_I$, $\tau \ll t_n$ забезпечується варіант розділового кола. В момент часу t_1 під час дії імпульсу (рис. 3.15) $U_R(t_1) = -U_C(t_1)$, а в момент часу t_2 мають місце нульові початкові умови. Сигнал на виході практично по-

второє сигнал на вході. Отже, таке коло є розділовим.

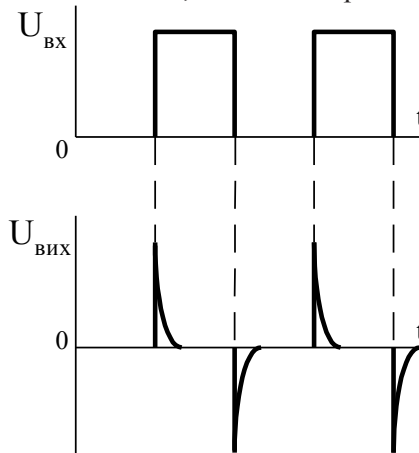


Рис. 3.14. Часові діаграми напруги на виході диференціювального кола при наявності імпульсної послідовності на вході

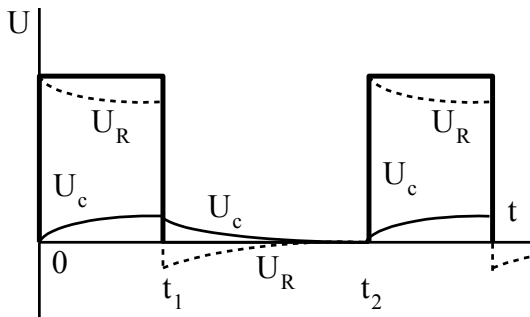


Рис. 3.15. Залежності $U_C(t)$ і $U_R(t)$; режим II при $\tau \gg t_I$, $\tau \ll t_n$

Інтегрувальні RC-кола.

Інтегрувальним колом називають чотириполіусник, сигнал на виході якого пропорційний інтегралу від вхідного сигналу. У випадку, коли вхідний і вихідний сигнали виражаються в однакових одиницях (наприклад, в одиницях напруги), операцію, виконувану інтегрувальним колом, можна записати у вигляді співвідношення

$$U_{\text{вих}}(t) = K \int_0^t U_{\text{вх}}(t) dt,$$

де K – коефіцієнт пропорційності, що має розрядність с^{-1} .

На рис. 3.16 наведена електрична схема інтегрувального кола. Припустимо, що $R_{\text{н}} \Rightarrow \infty$, що практично має місце в колі при $R_{\text{н}} \gg R_{\text{вих}}$.

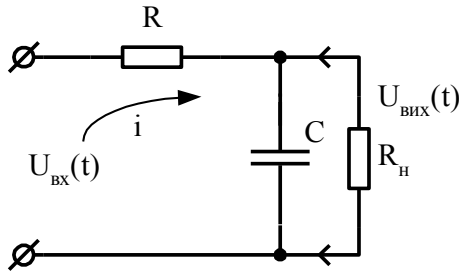


Рис. 3.16. Електрична схема інтегрувального RC-кола

Інтегрувальні кола часто застосовують для подовження (розширення) імпульсів або для отримання напруги, яка змінюється за законом, близьким до лінійного.

Для інтегрувального кола:

$$U_c = \frac{1}{C} \cdot \int_0^t i(t) dt,$$

а значення струму в колі $i(t) = \frac{U_{\text{вх}}(t) - U_{\text{вих}}(t)}{R}$.

Підставивши значення струму в формулу для U_c , отримаємо:

$$U_c = \frac{1}{C} \cdot \int_0^t \frac{U_{\text{вх}}(t) - U_{\text{вих}}(t)}{R} dt = \frac{1}{R \cdot C} \cdot \int_0^t [U_{\text{вх}}(t) - U_{\text{вих}}(t)] dt.$$

Для отримання ідеального інтегрувального кола необхідно, щоб виконувалася умова $U_{\text{вих}} \ll U_{\text{вх}}$, тоді запишемо:

$$U_c = \frac{1}{R \cdot C} \cdot \int_0^t U_{\text{вх}}(t) dt \cdot$$

Для того, щоб забезпечити низький коефіцієнт передачі кола, тобто $U_{\text{вих}} \ll U_{\text{вх}}$ ($U_{\text{вих}} = U_c$), необхідно забезпечити умову $\tau \gg t_I$ (рис. 3.17). Кут нахилу прямої на виході інтегрувального кола пропорційний амплітуді напруги імпульсу, що інтегрується, та обернено пропорційний сталій часу τ кола.

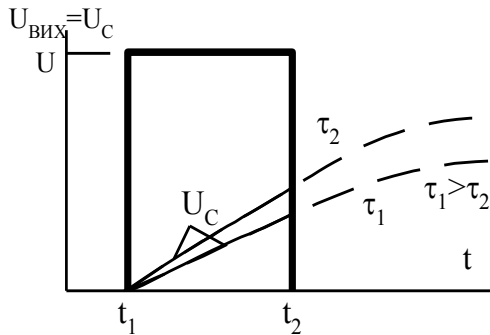


Рис. 3.17. Напруга на виході кола (на конденсаторі) при різних τ

Отже, для того, щоб коло забезпечувало функцію інтегрування, необхідно виконання таких умов:

$$U_{\text{вих}} \ll U_{\text{вх}}; \quad \tau \gg t_I.$$

Під час дії імпульсу ($t_1 \div t_2$) $U_{\text{вх}} = U = \text{const}$, тобто

$$U_{\text{вих}} = \frac{1}{R \cdot C} \cdot \int_0^t U \cdot dt = \frac{U}{R \cdot C} \cdot t \cdot$$

В момент часу t_2 напруга на виході кола буде дорівнювати

$$U_{\text{вих}}(t_2) = \frac{U}{R \cdot C} \cdot t_I \cdot$$

На рис. 3.18 показана вихідна напруга інтегрувального RC-кола, де позначено:

1 – реальна напруга інтегрувального кола $U_{\text{вих}} = U_c$;

2 – напруга на виході ідеального інтегратора.

При $\tau \gg t_I$ коло працює практично без похибки.

Визначимо максимальні амплітудні похибки, що мають місце наприкінці імпульсу:

$$\Delta_{\max} = \left. \frac{dU_c}{dt} \right|_{t=0} - \left. \frac{dU_c}{dt} \right|_{t=t_I}; \quad \delta_{\max} = \frac{\Delta_{\max}}{\left. \frac{dU_c}{dt} \right|_{t=0}},$$

де Δ_{\max} та δ_{\max} – максимальні значення абсолютної та відносної похибок.

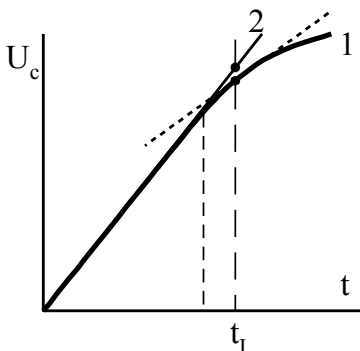


Рис. 3.18. До оцінки амплітудної похибки

Значення похибки при $t = t_I$ складе:

$$\delta_{t=t_I} = \frac{\left. \frac{dU_c}{dt} \right|_{t=0} - \left. \frac{dU_c}{dt} \right|_{t=t_I}}{\left. \frac{dU_c}{dt} \right|_{t=0}}.$$

3.2. Розрахунок лінійного формувача імпульсів

Розрахуємо електричне кола (рис. 3.19, а) на вхід якого подається сигнал $E(t)$ (рис. 3.19, б) з такими параметрами: $E = 5 \text{ В}$, $\tau_i = 100 \text{ мкс}$. Параметри кола мають такі значення: $R_1 = 800 \text{ Ом}$; $R_2 = 200 \text{ Ом}$; $C = 1 \text{ мкФ}$. В результаті розрахунку необхідно:

- визначити операторну передавальну функцію кола;
- визначити та побудувати перехідну характеристику кола;
- отримати аналітичне співвідношення вихідної напруги $U(t)$ в загальному вигляді для $t \geq t_1$;
- розрахувати сталу часу кола та визначити потрібні для побудови часової діаграми величини;
- побудувати відповідно до розрахованих даних часову діаграму $U(t)$, синхронну з $E(t)$;
- визначити тип кола (без врахування елементу R_2) та з'ясувати, з якою метою до схеми введено додатковий елемент R_2 .

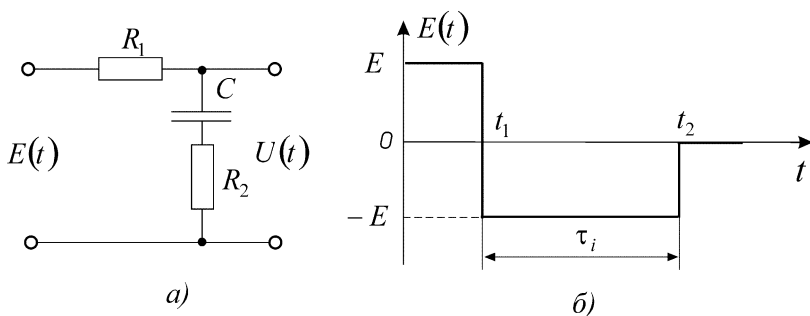


Рис. 3.19. Електрична схема лінійного формувача імпульсів (а) та часова діаграма його вхідного сигналу (б)

1.аходимо операторну передавальну функцію кола:

$$K(p) = \frac{R_2 + \frac{1}{pC}}{R_1 + R_2 + \frac{1}{pC}} = \frac{1 + pCR_2}{1 + pC(R_1 + R_2)} = \frac{R_2}{R_1 + R_2} \times$$

$$\times \frac{\frac{1}{R_2} + pC}{\frac{1}{R_1 + R_2} + pC} = \frac{R_2}{R_1 + R_2} \cdot \frac{1 + pCR_2}{\frac{R_2}{R_1 + R_2} + pCR_2} = \frac{K_R(1 + p\tau)}{K_R + p\tau},$$

де $K_R = \frac{R_2}{R_1 + R_2} < 1$; $\tau = CR_2$.

2. Перехідна характеристика кола $h(t)$ —це реакція кола на функцію включення (функцію Хевісайда) $1(t)$ при нульових початкових умовах (рис. 3.20, а), тобто

$$1(t) = \begin{cases} 0, & \text{при } t < 0; \\ 1, & \text{при } t \geq 0. \end{cases}$$

Знайдемо операторне зображення функції включення:

$$1(p) = \int_{-\infty}^{\infty} 1(t) e^{-pt} dt = \frac{1}{p}.$$

Тоді

$$h(p) = K(p)1(p) = \frac{K_R(1 + p\tau)}{p(K_R + p\tau)}. \quad (3.4)$$

Для знаходження перехідної характеристики як оригіналу функції $h(p)$ скористаємось зворотним перетворенням Лапласа. Якщо функція $h(p)$ представлена у вигляді раціональної

дроби $h(p) = \frac{R(p)}{Q(p)}$, то, якщо корені характеристичного рівняння

однократні, оригінал її можна знайти як суму лишків

$$h(t) = \sum_{i=1}^N \text{res}_i, \text{ при цьому}$$

$$\text{res}_i = \left. \frac{R(p)e^{pt}}{Q'(p)} \right|_{p=p_i},$$

де p_i — полюси функції $h(p)$, які є коренями рівняння $Q(p) = 0$.

Тоді

$$\begin{aligned}
 h(t) &= \sum_{i=2}^{N=2} \frac{K_R(1+p\tau)}{K_R+2p\tau} e^{p_i t} \bigg|_{\substack{p_1=0 \\ p_2=-\frac{K_R}{\tau}}} = \\
 &= 1 + \frac{K_R(1-K_R)}{K_R-2K_R} e^{-\frac{K_R}{\tau}t} = 1 - (1-K_R)e^{-\frac{K_R}{\tau}t}.
 \end{aligned} \tag{3.5}$$

За точками (при $t=0$ $h(0)=K_R$, при $t \rightarrow \infty$ $h(\infty) \rightarrow 1$) будемо функцію $h(t)$ (рис. 3.20, б).

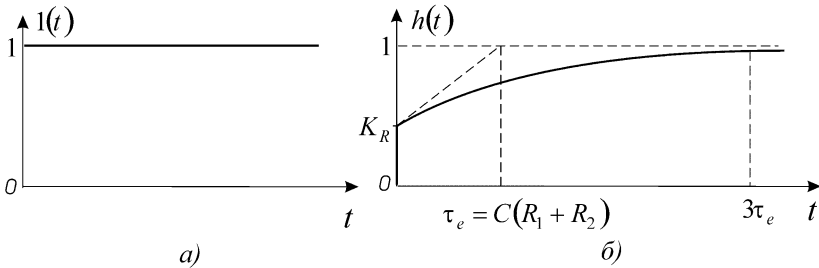


Рис. 3.20. Функція включення (а) та перехідна характеристика лінійного формувача імпульсів (б)

3. Знайти відгук електричного кола на задану вхідну імпульсну дію можна за допомогою трьох методів: класичного, операторного та формули експоненти. В зв'язку з тим, що в п.1 вже було знайдено операторну передавальну функцію електричного кола, знайдемо аналітичне співвідношення вихідної напруги $U(t)$ на інтервалі часу $t \geq t_1$ операторним методом. Для цього необхідно врахувати стрибок напруги в момент часу $t=t_1$ з амплітудою $-2E$ і операторним зображенням $E(p) = \frac{-2E}{p}$, та

додати до отриманого оригіналу значення E (початкову умову):

$$U(p) = K(p)E(p) + E = \frac{-2EK_R(1+p\tau)}{p(K_R + p\tau)} + E;$$

$$\begin{aligned}
U(t) &= E + \sum_{i=1}^{N=2} \frac{-2 E K_R (1 + p \tau)}{K_R + 2 p \tau} \bigg|_{\substack{p_1=0 \\ p_2=-\frac{K_R}{\tau}}} = \\
&= E - 2E + \frac{-2 E K_R (1 - K_R)}{K_R - 2 K_R} e^{-\frac{K_R}{\tau} t} = \\
&= -E + 2E (1 - K_R) e^{-\frac{K_R}{\tau} t} = E \left[2(1 - K_R) e^{-\frac{K_R}{\tau} t} - 1 \right]. \quad (3.6)
\end{aligned}$$

4. вихідними даними розраховуємо:

$$K_R = \frac{R_2}{R_1 + R_2} = \frac{200}{200 + 800} = 0.2;$$

$$\tau = C R_2 = 1 \cdot 10^{-6} \cdot 200 = 0.2 \cdot 10^{-3} \text{ с} = 0.2 \text{ мс};$$

$$\tau_e = \frac{\tau}{K_R} = C (R_1 + R_2) = 1 \cdot 10^{-6} \cdot (200 + 800) = 1 \cdot 10^{-3} = 1 \text{ мс}.$$

Підставивши в співвідношення (3.6) $t = t_1 = 0$, знаходимо вихідну напругу в момент t_1^+ після стрибка

$$\begin{aligned}
U(t_1^+) &= U(0) = E [2(1 - K_R) - 1] = E (1 - 2 K_R) = \\
&= 5 \cdot (1 - 2 \cdot 0.2) = 3 \text{ В},
\end{aligned}$$

а підставивши в співвідношення (3.6) $t = t_2 = \tau_i$, отримаємо напругу в момент t_2 до стрибка вхідного сигналу

$$\begin{aligned}
U(t_2) &= E \left[2(1 - K_R) e^{-\frac{K_R}{\tau} \tau_i} - 1 \right] = \\
&= 5 \left[2(1 - 0.2) e^{-\frac{0.2 \cdot 100}{200}} - 1 \right] = 2.24 \text{ В}.
\end{aligned}$$

Після стрибка вхідної напруги $E(t)$ в момент t_2^+ на ви-

хід кола передається позитивний стрибок напруги $\Delta U = |E| \cdot K_R = 5 \cdot 0.2 = 1 \text{ В}$, тому

$$U(t_2^+) = U(t_2) + \Delta U = 2.24 + 1 = 3.34 \text{ В}.$$

5. розрахованими даними будуємо часову діаграму $U(t)$, синхронну з $E(t)$ (рис. 3.21). При цьому враховуємо, що при $\tau_e \gg \tau_i$ на інтервалі τ_i напруга $U(t)$ змінюється майже лінійно, а при $t > t_2$ конденсатор розряджається до нуля і напруга $U(t)$ експоненційно прямує до нуля з меншою швидкістю.

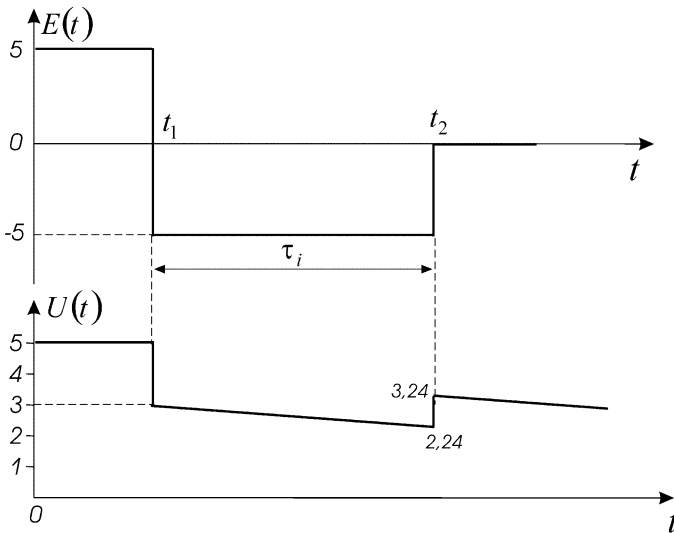


Рис. 3.21. Часові діаграми вхідного $E(t)$ та вихідного $U(t)$ сигналів лінійного формувача імпульсів

6. Наведене на рис. 3.19, а коло за умови $R_2 = 0$ є інтегровальним, тому що стала часу кола $\tau_e = 1 \text{ мс}$ набагато більше тривалості імпульсу $\tau_i = 0.1 \text{ мс}$. Резистор R_2 включений послідовно з конденсатором C для створення стрибків вихідної напруги в моменти комутації вхідного сигналу (рис. 3.21). Зав-

дяки цьому замість імпульсів лінійно-змінної напруги коло формує імпульси трапецевидної форми.

3.3. Обмежувачі послідовного і паралельного типу на діодах

Обмежувачі напруги призначені для захисту вхідних кіл електронних пристроїв від небезпечного для них рівня вхідних сигналів. Крім того, в деяких випадках вони можуть використовуватись для формування імпульсів.

На рис. 3.22 наведена структурна схема захисту електронних пристроїв за допомогою обмежувача. Вхідна напруга перевищує допустиме значення на вході обмежувача, а на його виході рівень напруги знаходиться у допустимих межах.

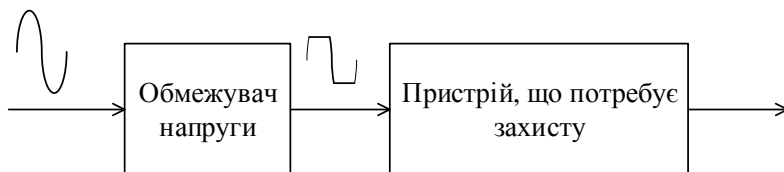


Рис. 3.22. Структурна схема захисту електронних пристроїв за допомогою обмежувача

Розрізняють обмежувачі зверху, знизу і двосторонні обмежувачі. Амплітудні характеристики обмежувачів і осцилограми, що ілюструють їх роботу, наведені на рис. 3.23 ÷ рис. 3.25.

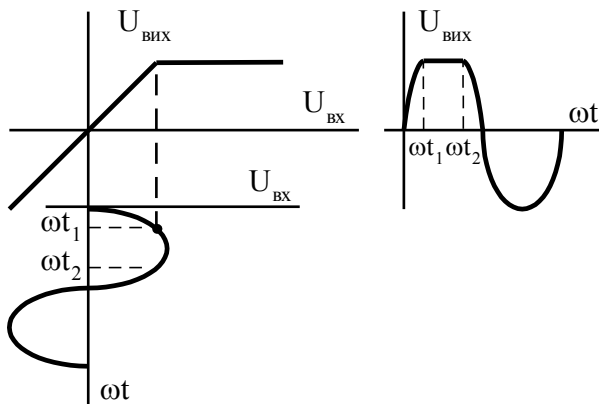


Рис. 3.23. Діаграми, що пояснюють роботу одностороннього обмежувача зверху

Послідовні обмежувачі на діодах.

Електрична схема послідовного обмежувача знизу на приблизно нульовому рівні наведена на рис. 3.26 разом з осцилограмою, що ілюструє її роботу.

Опір навантаження має бути набагато більше опору обмежувача R ($R_H \gg R$), а внутрішній опір джерела синусоїдального сигналу $R_{\text{вн}} \ll R$. Для забезпечення протікання струму в інтервалі від 0 до π $U_{\text{вх.м}} \gg U_{\text{д.пр}}$, де $U_{\text{д.пр}}$ – напруга на діоді, включеному у прямому напрямку, яка має порядок $(0,3 \dots 0,7)$ В.

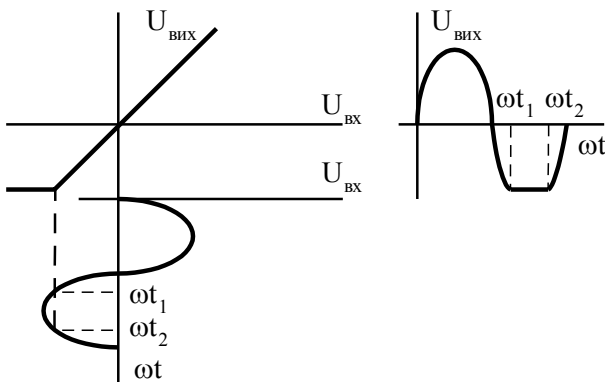


Рис. 3.24. Діаграми, що пояснюють роботу одностороннього обмежувача знизу

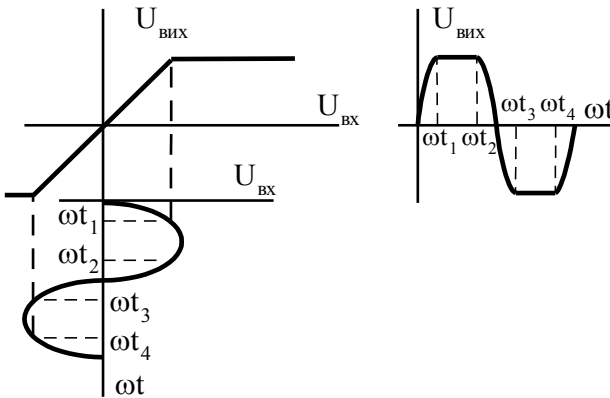


Рис. 3.25. Діаграми, що пояснюють роботу двостороннього обмежувача

Схема обмежувача *зверху* на приблизно нульовому рівні та осцилограми його роботи аналогічні і мають вигляд, наведений на рис. 3.27.

Для того, щоб схеми обмежувачів були універсальними, вони мають забезпечувати обмеження на довільному рівні. Схема обмежувача *зверху* на довільному рівні наведена на рис. 3.28. Полярність джерела ЕРС U_{on} вибирають так, щоб діод VD був відкритий при $E_{ex} = 0$. Опорна напруга U_{on} може змінюватися в межах від 0 до $E_{ex\ max}$. Оскільки $R_{вн}$ і $R_{VD\ np}$ набагато менші від величини резистора R , можна зробити припущення, що $R_{вн} \approx 0$; $R_{VD\ np} \approx 0$.

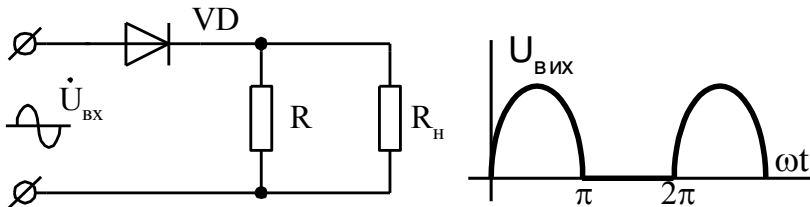


Рис. 3.26. Послідовний обмежувач на діоді

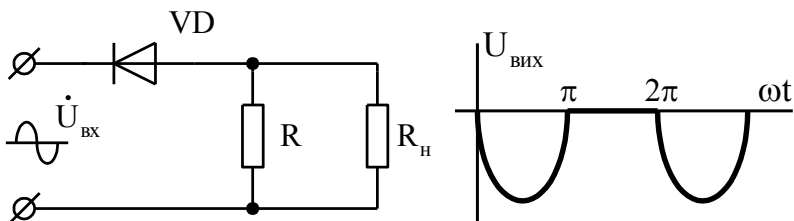


Рис. 3.27. Обмежувач зверху на нульовому рівні

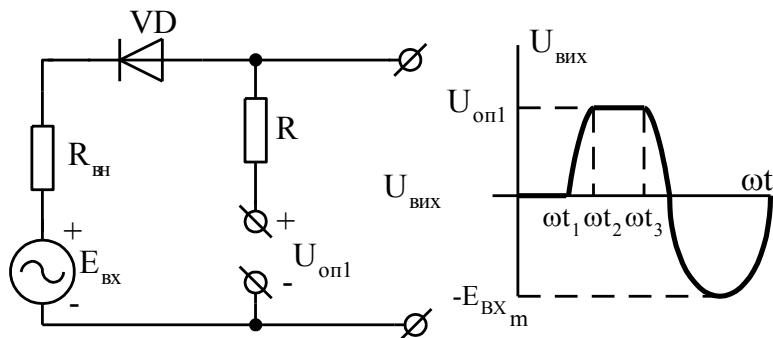


Рис. 3.28. Обмежувач довільного рівня зверху

При виконанні умови $E_{\text{вх}} \leq U_{\text{оп}}$ діод VD відкритий, а при виконанні зроблених припущень $U_{\text{вих}} = E_{\text{вх}}$. Якщо $E_{\text{вх}} > U_{\text{оп}}$, то діод VD закритий: струм в контурі (і через резистор R) дорівнює нулю, отже $U_{\text{вих}} = U_{\text{оп}}$ (інтервал аргументу $\omega t_2 \div \omega t_3$). На інтервалі від 0 до ωt_1 при $E_{\text{вх}} = 0$ отримують $U_{\text{вих}} = 0$. Схема обмежувача знизу на довільному рівні наведена на рис. 3.29. Робота схеми, часова діаграма та припущення аналогічні розглянутим вище.

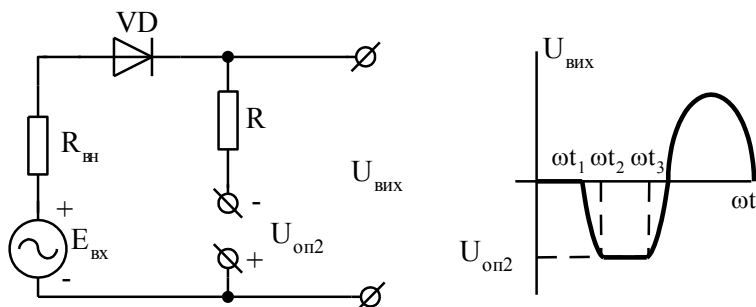


Рис. 3.29. Обмежувач знизу на довільному рівні

Схема обмежувача рівня зверху і знизу на довільних рівнях є комбінацією двох розглянутих схем, включених послідовно (рис. 3.30).

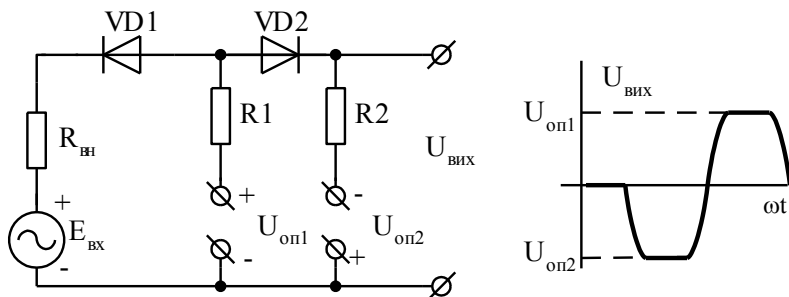


Рис. 3.30. Обмежувач зверху і знизу на заданих рівнях

Недоліками послідовних обмежувачів на діодах є:

- обмежувачам необхідні джерела ЕРС вхідного сигналу з параметрами, близькими до ідеальних ($R_{вн} \approx 0$).
- схеми є пасивними і мають коефіцієнт передачі $K < 1$.
- амплітуда вхідної ЕРС $E_{вх}$ має бути великою (десятки вольт) для виконання умови $E_{вх} \gg U_{д.пр}$.

Паралельні обмежувачі на діодах.

Основним недоліком послідовних обмежувачів на діодах є потреба у низькому внутрішньому опорі джерела сигналу. Для виключення цього недоліку розроблені паралельні обмежувачі на діодах. Такі схеми обмежувачів не вимагають дуже низького

вихідного опору джерела ЕРС. Опір навантаження, як і у послідовних обмежувачів, має бути $R_H \gg R$.

Схема обмежувача на діодах *зверху* приблизно на нульовому рівні та часові діаграми, що ілюструють її роботу, наведені на рис. 3.31. Приймаючи припущення $R_H \gg R$ і $E_{ex} \gg U_{д.нр}$, опишемо схему системою рівнянь:

$$\begin{cases} E_{BX} = I \cdot R + U_{VD}; \\ U_{VD} = \varphi(I). \end{cases}$$

Рівняння перше – навантажувальна пряма 1, друге – вольт-амперна характеристика діода 2. Побудувавши навантажувальну пряму за точками холостого ходу ($I = 0$, $U_{\partial} = E_{ex}$) і короткого замикання ($U_{\partial} = 0$, $I_{кз} = \frac{E_{ex}}{R}$), побудуємо часову діаграму $U_{\partial}(\omega t) = U_{вих}$ при синусоїдальній вхідній ЕРС E_{ex} і отримаємо обмеження зверху на рівні $U_{д.нр} \approx (0.3 \div 0.6) B$.

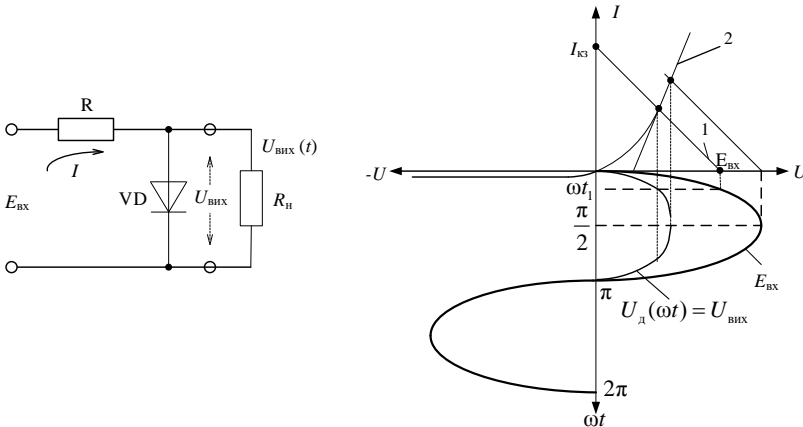


Рис. 3.31. Паралельний обмежувач зверху

Паралельний обмежувач на діодах *знизу* на приблизно нульовому рівні відрізняється полярністю підключення діода, а для обмеження на довільних рівнях схеми доповнюються джерелами напруги U_{on} , полярності яких вибирають так, щоб діоди були закриті при $E_{ex} = 0$.

Контрольні питання

1. Намалюйте схему RC-кола і наведіть умови, при яких коло буде диференціювальним.
2. За яких умов RC-коло буде розділовим?
3. Назвіть основні джерела похибок в інтегрувальних колах та наведіть співвідношення для їх визначення.
4. Назвіть переваги інтеграторів на операційних підсилювачах.
5. У чому перевага паралельних діодних обмежувачів відносно послідовних? Назвіть основні недоліки діодних обмежувачів.

4. ТРАНЗИСТОРНІ НАСИЧЕНІ КЛЮЧІ

4.1. Призначення транзисторних ключів

Електронні ключі (ЕК) використовують в імпульсній та

цифровій техніці для комутації сигналів в різних електричних колах. Найчастіше ключові схеми виконують на транзисторах, діодах або тиристорах. Розрізняють два види ключів:

- транзисторні насичені ключі;
- аналогові ключі.

Для передачі або відключення аналогових сигналів в радіоелектронній апаратурі застосовують електронні аналогові ключі, які при подачі сигналу управління (найчастіше сигналу логічної одиниці) переходять в замкнений або розімкнений стан. При замиканні ключа його внутрішній опір зменшується до значень $R_{\text{замкн.}} = (0.3 \dots 300) \text{ Ом}$ в ключах на біполярних транзисторах (БТ) і до значень $0,0003 \text{ Ом}$ в кращих ключах на польових (уніполярних) транзисторах (ПТ). При цьому в закритому стані вони мають опір близько $R_{\text{розімкн.}} = (50 \dots 500) \text{ кОм}$ для діодних ключів і ключів на БТ та порядку $(1 \dots 10) \text{ МОм}$ для ключів на ПТ. У кращих ключах на ПТ з ізольованим затвором опір розімкненого ключа може досягати 10^{12} Ом .

Транзисторні насичені ключі використовуються для формування імпульсних сигналів різної амплітуди, для перетворення рівнів імпульсних сигналів і керування різними пристроями (наприклад, електромагнітними реле або світлодіодними індикаторами).

4.2. Лінійні моделі транзисторів в режимі великого сигналу

На відміну від режиму малого сигналу, де відхилення від робочої точки за постійним струмом не перевищує $(20 \dots 30)\%$, в режимі великого сигналу транзистор швидко переходить із зони відсікання через активну область в режимі насичення та навпаки. Як правило, в імпульсній техніці транзистор працює в двох протилежних станах: в режимі відсікання (транзистор закритий) та в режимі насичення (транзистор відкритий і насичений). Коefіцієнт передачі транзистора в цих режимах менше одиниці, тобто він не має підсилювальних властивостей.

Крім того, при переході з одного режиму в другий та навпаки транзистор знаходиться в активному режимі, а час пере-

микання складає одиниці мікросекунд. У перехідному (активно-му) режимі коефіцієнт передачі транзистора набагато більше одиниці. В режимі великого сигналу характеристики транзистора нелінійні і принцип накладення не може бути застосований.

Для аналізу схем з транзисторами, що працюють в режимі великого сигналу, застосовують методи аналізу нелінійних схем. В інженерній практиці часто використовується метод апроксимації нелінійної ВАХ кусково-лінійними функціями.

Сутність методу полягає в тому, що для окремих областей (відсікання, насичення, перехідна область) виконується апроксимація нелінійних ВАХ кусково-лінійними функціями. У кожній області на основі апроксимуючих функцій ВАХ представляються рядом Тейлора. Однак всіма похідними, починаючи з другої, можна знехтувати, а ряд обмежитися двома доданками типу $a + b \cdot x$ (постійну складову необхідно врахувати).

На основі отриманих рівнянь для кожної з областей з урахуванням постійних складових синтезують електричну модель транзистора. При цьому моделі виходять лінійними для всіх трьох областей, однак з різними коефіцієнтами. Часто використовуються моделі транзисторів у системах h -, Y - та Z -параметрів.

Розглянемо апроксимацію вхідних і вихідних характеристик біполярного транзистора (схема з СЕ) і польового транзистора (з індукованим каналом). На рис. 4.1 представлені вихідні характеристики біполярного транзистора.

Область відсікання (1) розташована між характеристиками $I_{\bar{b}} = 0$ і $I_{\bar{b}} = -I_{\kappa 0}$ з відповідними значеннями струмів колектора $I_{\kappa \text{ поч}}$ і $I_{\kappa 0}$. Область насичення (3) відповідає мінімальним значенням напруги U_{KE} . Опір транзистора в режимі насичення $R_{KE, \text{нас}}$ визначається тангенсом кута нахилу лінії 3, тобто $\text{tg} \beta$. Між ними знаходиться область активного режиму (2). На вхідних характеристиках транзистора (рис. 4.2) також зазначені ці три основні області.

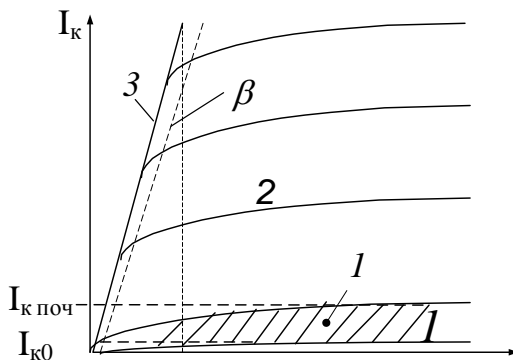


Рис. 4.1. Вихідні характеристики біполярного транзистора

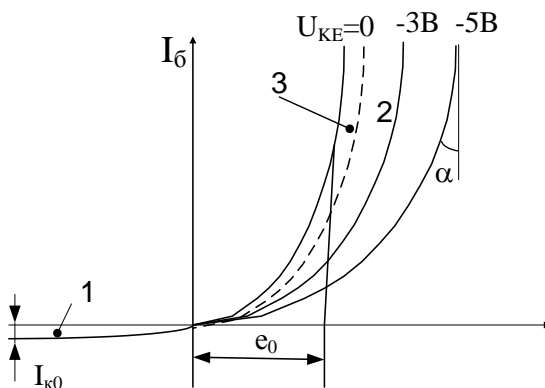


Рис. 4.2. Вхідні характеристики біполярного транзистора

Лінійна апроксимація вхідних та вихідних характеристик біполярного транзистора з виділенням характерних областей наведена на рис. 4.3.

Характеристики польового транзистора з індукованим каналом аналогічні розглянутим, а їх апроксимація для трьох областей (рис. 4.4) аналогічна.

Встановимо значення струмів колектора для транзистора, що знаходиться в режимі відсікання. Для схеми з спільною базою при $I_E = 0$ між базою і колектором протікає зворотний (тепловий) струм величиною I_{K0} (рис. 4.5, а). Для схеми з спільним емітером для забезпечення $I_B = 0$ необхідно через пере-

хід емітер-база пропускати струм I_{K0} (рис. 4.5, б). При цьому струм колектора буде $I_{K\text{ поч}} = I_{K0} \cdot h_{21E}$, тобто в h_{21E} разів більше, ніж для схеми з спільною базою. Зменшити струм $I_{K\text{ поч}}$ до величини I_{K0} можна шляхом закривання транзистора потенціалом відсікання на базу, пропускаючи струм з бази в емітер величиною I_{K0} , в результаті чого $I_E = 0$, а $I_K = I_{K0}$ (рис. 4.5, в).

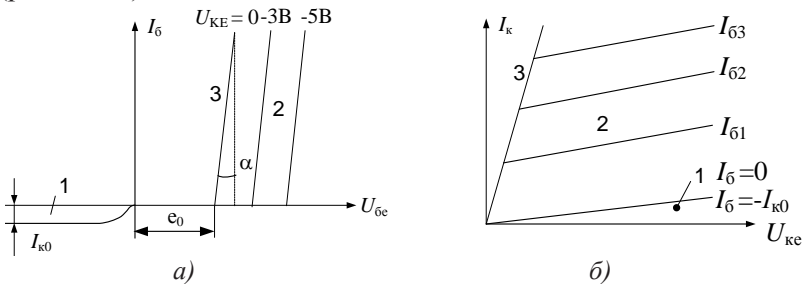


Рис. 4.3. Апроксимація входних (а) та вихідних (б) характеристик біполярного транзистора

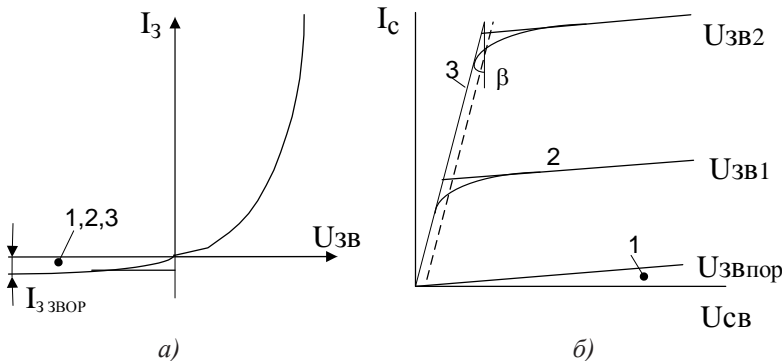


Рис. 4.4. Входні (а) та вихідні (б) характеристики польового транзистора з індукованим каналом

Електрична модель транзистора для області активного режиму (2) була розглянута раніше.

Модель транзистора в області відсікання ($I_E = 0$), що забезпечується подачею позитивного потенціалу на базу віднос-

но емітера (спільної шини), має вигляд, наведений на рис. 4.6.

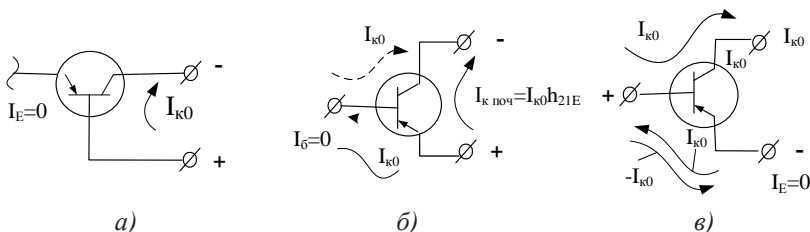


Рис. 4.5. Струми колектора транзисторів в режимі відсікання

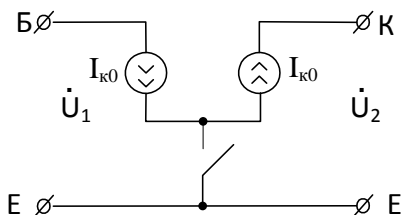


Рис. 4.6. Електрична модель транзистора для області відсікання (1)

Для області насичення (3) модель транзистора має вигляд (рис. 4.7), де e_0 визначається при апроксимації вхідної характеристики. При цьому $e_0 = U_{BE \text{ нас}} \approx 0,2 \div 0,5B$, $h_{11E} = tg \alpha$ (рис. 4.3, а), $R_{K \text{ нас}} = tg \beta$ (рис. 4.1).

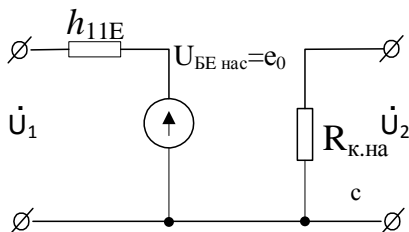


Рис. 4.7. Електрична модель транзистора в області насичення (3)

Модель транзистора в області відсікання являє собою розрив емітера відносно колекторно-базового переходу, через який протікає струм I_{K0} . Для цього режиму необхідно подати на базу транзистора потенціал відсікання відносно емітера.

Модель транзистора в режимі насичення являє собою

практично замкнені електроди база, колектор і емітер, тому що $R_{к\text{ нас}} \approx (1 \div 10) \text{ Ом}$; $e_0 \approx (0,2 \div 0,5) \text{ В}$; $h_{11E} \approx (10 \div 100) \text{ Ом}$.

Для забезпечення режиму насичення транзистора необхідно подати на базу струм, значно більший струму в режимі відсікання.

4.3. Методика розрахунку параметрів транзисторного ключа

Розрахунок транзисторного ключа в режимі насичення. Електрична схема ключа наведена на рис. 4.8. Для насичення ключа (рис. 4.8, а) необхідні позитивні входні імпульси визначеної амплітуди. В результаті розрахунків потрібно визначити тип транзистора, тобто параметри E_K , R_K , $R_B(E_I)$. Для негативних входних імпульсів застосовують транзистори провідності типу p-n-p (рис. 4.8, б).

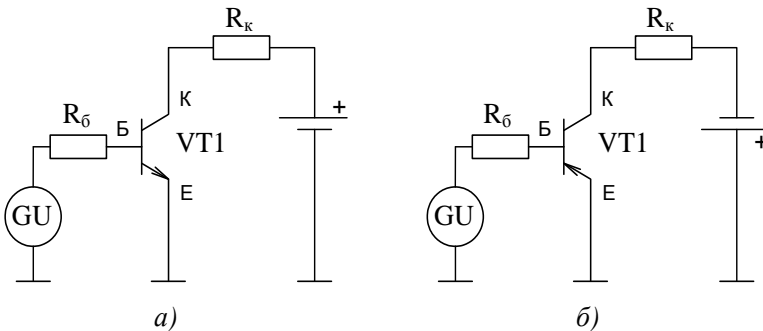


Рис. 4.8. Схеми включення біполярних транзисторів провідностей типу n-p-n (а) та p-n-p (б)

Розрахунок схеми за постійним струмом традиційний, однак враховується входна характеристика транзистора в режимі насичення ($U_{KE} \approx 0$):

$$\begin{cases} E_K = U_{KE.нас} + I_K R_K; \\ U_{KE.нас} = I_K R_{K.нас}. \end{cases}$$

Навантажувальна лінія (1) будується за двома точками (режими ХХ і КЗ) (рис. 4.9):

$$XX: I_K = 0, U_{KE} = E_K;$$

$$K3: U_{KE} = 0, I_{K.3.} = \frac{E_K}{R_K}.$$

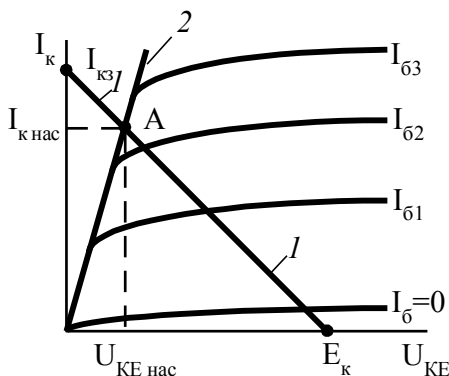


Рис. 4.9. До розрахунку ключа за постійним струмом

Перетином навантажувальної прямої (1) з лінією насичення (2) є точка "А", що визначає режим насичення та його параметри $I_{K \text{ нас}}$, $U_{KE \text{ нас}}$ та $I_{B \text{ нас}}$, в даному випадку це I_{B_3} (рис. 4.9). При проектуванні ключів струм колектора в режимі насичення зазвичай заданий, що визначає вибір типу транзистора за припустимим струмом і положенню точки "А" (тобто, і струму $I_{K.3.}$). За значенням цього струму розраховують резистор R_K за формулою $R_K = \frac{E_K}{I_{K.3.}}$.

Для розрахунку резистора R_{δ} скористаємося вхідною характеристикою транзистора в режимі насичення ($U_{KE} \approx 0$). Струм бази, зумовлений E_I та резистором R_{δ} , має бути: $I_{\delta} \geq I_{\delta_3}$. Встановимо положення точки "А" на вхідній характеристиці за значенням струму бази в точці "А" на вихідних характеристиках для I_{δ_3} (рис. 4.10). Якщо E_I задано, то навантажувальна пряма до вхідних характеристик має перейти з точки XX (E_I) через точку "А" визначити значення струму КЗ

($I_{\bar{\sigma} \text{ к.з.}}$). Оскільки струм $I_{\bar{\sigma} \text{ к.з.}} = E_I / R_{\bar{\sigma}}$, то звідси можна визначити опір резистора $R_{\bar{\sigma}}$ як $R_{\bar{\sigma}} = E_I / I_{\bar{\sigma} \text{ к.з.}}$.

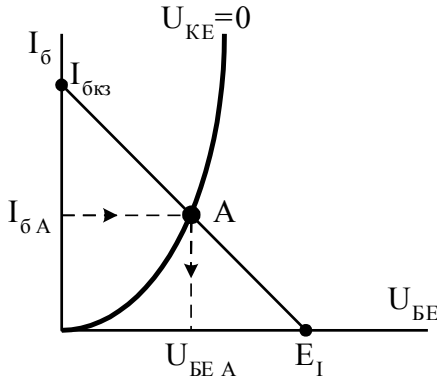


Рис. 4.10. Визначення положення робочої точки на входних характеристиках

Для прискорення процесу насичення ключа вводять поняття коефіцієнта насичення: $S = \frac{I_{\bar{\sigma}}}{I_{\bar{\sigma} \text{ нас}}}$, $I_{\bar{\sigma} \text{ нас}} = I_{\bar{\sigma} A}$.

Рекомендоване значення коефіцієнта $S \cong 1,5 \div 2$, тому що при збільшенні S зменшується час вмикання ключа, однак при цьому збільшується час вимикання.

Найпростіший розрахунок ключа.

$I_{\text{к нас}}$ – координата точки "А«» $E_{\text{к}}$ – напруга, задана споживачем. Наприклад: $I_{\text{к нас}} = 0,1 \text{ А}$; $E_{\text{к}} \approx 10 \text{ В}$.

Прийmemo, що:

$$I_{\text{к.з.}} \cong I_{\text{к нас}} \Rightarrow R_{\text{к}} = \frac{E_{\text{к}}}{I_{\text{к нас}}} = \frac{10}{0,1} = 100 \text{ Ом},$$

тоді з врахуванням середнього коефіцієнта підсилення транзистора за струмом $h_{21E} \approx 50$ розрахуємо струм бази насичення:

$$I_{\bar{o} \text{ нас}} = \frac{I_{\kappa \text{ нас}}}{h_{21E}} = \frac{0,1}{50} \approx 2 \text{ мА}.$$

Прийmemo $S = 2$, тоді $I_{\bar{o}} = S \cdot I_{\bar{o} \text{ нас}} = 4 \text{ мА}$. Далі розрахуємо $R_{\bar{o}}$ при заданому значенні амплітуди імпульсу, наприклад, при $E_I = 5 \text{ В}$, $U_{BE.A} \approx 0,4 \text{ В}$ отримаємо:

$$R_{\bar{o}} = \frac{E_I - U_{BE.A}}{I_{\bar{o}.A}} = \frac{5 - 0,4}{4 \cdot 10^{-3}} = 1,1 \text{ кОм}.$$

Орієнтовні вимоги до транзистора:

$$U_{KE \text{ проб}} = 15 \div 20 \text{ В}; \quad I_{\kappa \text{ max}} = 150 \div 200 \text{ мА}; \quad h_{21E} \approx 50.$$

Повна модель ключа для області насичення наведена на рис. 4.11. При цьому $R_{\kappa \text{ нас}} \ll R_{\kappa}$, що забезпечує $U_{KE \text{ нас}} \approx 0$.

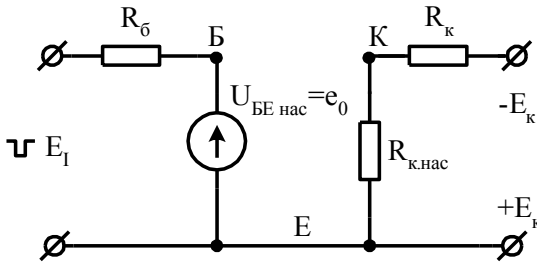


Рис. 4.11. Повна модель ключа для області насичення при $I_{\bar{o}} \geq I_{\bar{o} \text{ нас}}$

Спрощена модель ключа наведена на рис. 4.12. У спрощеному варіанті можна вважати, що виводи транзистора К, Е та Б мають однаковий потенціал.

Розрахунок транзисторного ключа в режимі відсікання. Схема ключа і вихідні дані для розрахунку за постійним струмом в даному режимі наведені на рис. 4.13.

На вхід схеми подаються позитивні імпульси, що замикають транзистор. Робоча точка A_1 для повністю закритого транзистора має розташовуватися на самій нижній характеристиці ($I_{\bar{o}} = -I_{\kappa 0}$). При цьому $U_{KE \text{ відс}} \approx -E_{\kappa}$. Розрахунок режиму за постійним струмом аналогічний. Модель ключа в режимі відсі-

кання наведена на рис. 4.14.

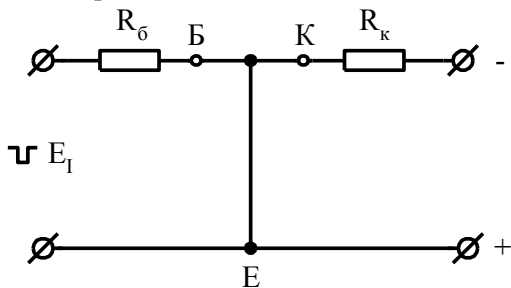


Рис. 4.12. Спрощена модель ключа при $I_{\delta} \geq I_{\delta \text{ нас}}$

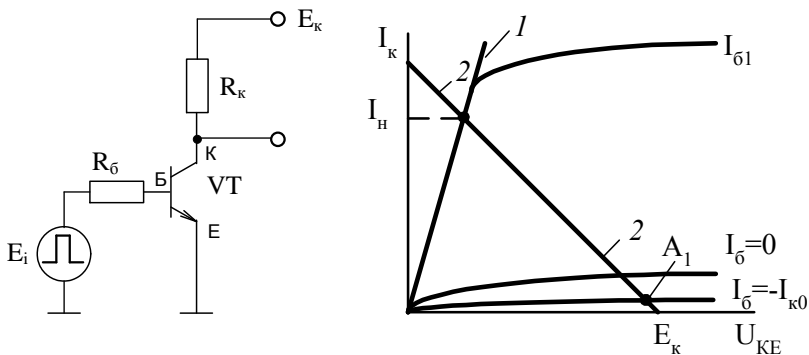


Рис. 4.13. Схема ключа в режимі відсікання та вихідні дані для розрахунків за постійним струмом

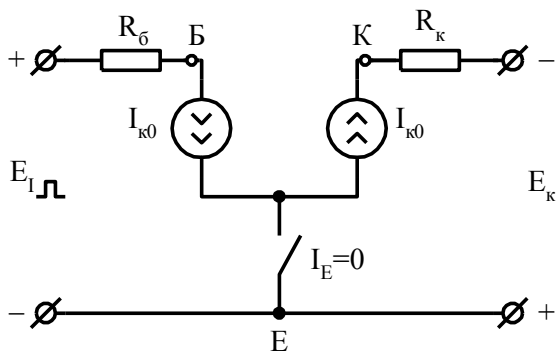


Рис. 4.14. Модель ключа в режимі відсікання

Розрахуємо необхідну амплітуду імпульсів (E_I), що подаються на вхід ключа і забезпечують закривання транзистора (точка A_1).

За відомим описом вхідна характеристика транзистора

$$I_{\bar{\sigma}} = I_{\bar{\sigma} \text{ нас}} \cdot \left(e^{\frac{-U_{BE}}{\varphi_T}} - 1 \right),$$

де $I_{\bar{\sigma} \text{ нас}}$ – значення струму бази при зворотному включенні вхідного переходу транзистора, в режимі відсікання $I_{\bar{\sigma} \text{ нас}} = -I_{\kappa 0}$; φ_T – тепловий потенціал (25 мВ при $T = 293^\circ\text{K}$).

Тоді

$$I_{\bar{\sigma}} = I_{\kappa 0} \cdot \left(e^{\frac{-U_{BE}}{\varphi_T}} - 1 \right).$$

Якщо $e^{\frac{-U_{BE}}{\varphi_T}} \ll 1$, то $I_{\bar{\sigma}} = -I_{\kappa 0}$, що забезпечує роботу

транзисторного ключа в точці A_1 . Для забезпечення $e^{\frac{-U_{BE}}{\varphi_T}} \ll 1$ необхідно, щоб $U_{BE} \approx 10\varphi_T = 250 \text{ мВ}$.

Знайдемо амплітуду вхідних імпульсів:

$$E_I = U_{BE \text{ відс}_{\max}} + I_{\kappa 0} \cdot R_{\bar{\sigma}}.$$

Значення напруги $I_{\kappa 0} \cdot R_{\bar{\sigma}}$ має порядок десятків мілівольт, тому с запасом $E_I \approx (0,3 \div 0,5) \text{ В}$. Тому спрощена модель ключа є розімкненими колектором і емітером при напрузі на базі $(0,3 \div 0,5) \text{ В}$.

4.4. Аналіз перехідних процесів в транзисторному ключі

Розглянемо випадок, якщо на вхід транзисторного ключа

(рис. 4.15, а) подається сигнал $E(t)$ (рис. 4.15, б) з параметрами $E^1 = 2.4 \text{ В}$ та $E^0 = 0.4 \text{ В}$. Потрібні параметри схеми та вихідного сигналу транзисторного ключа: допустима тривалість фронту вихідної напруги $t_\phi = 0.5 \text{ мкс}$; рівень логічної одиниці вихідної напруги $|U^1| \geq 9.5 \text{ В}$; рівень логічного нуля вихідної напруги $|U^0| \leq 1 \text{ В}$; максимально допустимий струм, що споживається схемою від джерела входного сигналу $I_{ex.max} = 1 \text{ мА}$; завадостійкість $|U_3| \geq 0.5 \text{ В}$. Необхідно:

1) знайти операторну передавальну функцію входного струму, перехідну характеристику транзисторного ключа та в загальному випадку отримати аналітичне співвідношення для уявного струму колектора $i_k(t)$ при $t \geq t_1$;

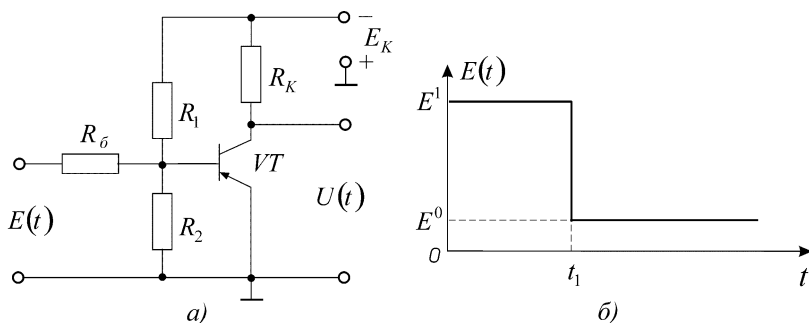


Рис. 4.15. Схема транзисторного ключа (а) та часова діаграма його входного сигналу (б)

2) провести частковий ескізний розрахунок транзисторного ключа: вибрати тип транзистора, вибрати напругу джерела живлення та розрахувати параметри елементів схеми;

3) розрахувати потрібні для побудови часових діаграм величини: напруги та струми в характерних точках часових діаграм, а також тривалості етапів перехідного процесу;

4) відповідно до розрахованих даних побудувати часові діаграми $i_b(t)$, $U_b(t)$, $i_K(t)$ та $U_K(t)$, синхронні з $E(t)$;

5) вказати, як зміняться розраховані в п. 3 величини при збільшенні параметру E^1 ; результати пояснити, виходячи з фізичних міркувань.

Розрахунок транзисторного ключа.

1. вхід схеми транзисторного ключа при $t < t_1$ подається позитивна напруга високого рівня $E^1 = 2.4 B$, а при $t \geq t_1$ – низького рівня $E^0 = 0.4 B$. Таким чином, транзисторний ключ на транзисторі $p-n-p$ типу може переключатися, якщо при $E = E^1$ транзистор закритий, а при $E = E^0$ – відкритий. Якісно фізичні процеси при включенні транзисторного ключа протікають таким чином. У вихідному стані $t < t_1$ транзистор закритий, напруга на його базі U_{ϕ}^+ має бути більш позитивною, ніж порогове значення $U_{\phi 0}$, наприклад, $U_{\phi}^+ \geq 0$. Вхідна паразитна ємність C_{BX} заряджена до цієї напруги, тому при $t \geq t_1$ за час перезаряду конденсатора C_{BX} спочатку виникає затримка включення, а потім починається включення.

Покладаючи для простоти $t_1 = 0$ та враховуючи, що $i_K(0) = 0$, за формулою експоненти $f(t) = f(\infty) - \Delta f e^{-t/\tau}$ отримуємо співвідношення для колекторного струму як функцію часу

$$i_K(t) = \beta I_{\phi}^1 \left(1 - e^{-t/\tau_{\beta}} \right), \quad (4.1)$$

де $\tau_{\beta} = \frac{1}{\omega_{\beta}} = \frac{1}{2\pi f_{\beta}}$ – стала часу, яка характеризує інерційність транзистора та дорівнює часу життя носіїв в базі; f_{β} – гранична частота транзистора, на якій $\beta = 0.707 \beta_0$; β_0 – коефіцієнт передачі струму бази на низьких частотах; f_{α} – гра-

нича частота, на якій $\alpha = 0.707 \alpha_0$; α_0 – коефіцієнт передачі струму емітера на низьких частотах; $\tau_\alpha = \frac{1}{\omega_\alpha} = \frac{1}{2\pi f_\alpha} = \frac{1}{2\pi f_T}$; f_T – частота, на якій $|\beta| = 1$; $\tau_\alpha = \frac{\tau_\beta}{1 + \beta} \approx \frac{\tau_\beta}{\beta}$.

Оскільки I_σ^1 – величина стала, можна вважати, що за час включення коефіцієнт передачі транзистора змінюється за законом

$$h_\beta(t) = \beta \left(1 - e^{-t/\tau_\beta} \right), \quad (4.2)$$

де $h_\beta(t)$ – перехідна характеристика транзистора.

Враховуючи, що $L\{1(t)\} = \frac{1}{p}$ та $L\{e^{\alpha t}\} = \frac{1}{p - \alpha}$, перепишемо співвідношення для колекторного струму в операторній формі:

$$\begin{aligned} I_K(p) &= \beta I_\sigma^1 \left(\frac{1}{p} - \frac{1}{p + \frac{1}{\tau_\beta}} \right) = \beta I_\sigma^1 \frac{p + \frac{1}{\tau_\beta} - p}{p \left(p + \frac{1}{\tau_\beta} \right)} = \\ &= \frac{\beta I_\sigma^1}{\tau_\beta} \cdot \frac{1}{p \left(p + \frac{1}{\tau_\beta} \right)} = \frac{I_\sigma^1}{p} \cdot \frac{\beta}{1 + p \tau_\beta}. \end{aligned} \quad (4.3)$$

Колекторний струм можна представити як добуток базового струму на передавальну функцію:

$$I_K(p) = I_\sigma(p) K(p). \quad (4.4)$$

Оскільки $I_\sigma(p) = \frac{I_\sigma^1}{p}$, то з виразу (4.3) отримаємо

$$K(p) = \frac{I_K(p)}{I_{\sigma}(p)} = \frac{\beta}{1 + p\tau_{\beta}}, \quad (4.5)$$

що i є операторною передавальною функцією транзисторного ключа.

При цьому до того моменту, коли струм колектора досягне значення $I_{K.H}$, експонента в співвідношенні (4.1) для $i_K(t)$ з сталою часу $\tau_{\beta \text{ екс}} = \tau_{\beta} + \beta C_{K\sigma} R_K$ описує реальний колекторний струм транзистора, а далі з сталою часу $\tau_H \approx \tau_{\beta}$ – уявний струм, де $C_{K\sigma}$ – паразитна ємність переходу колектор – база, а τ_H – стала часу транзистора в насиченому стані.

2. Оскільки задано $|U^1| \geq 9.5 \text{ В}$, виходячи з того, що

$$U^1 = -(E_K - I_{K0} R_K),$$

вибираємо $E_K = -18 \text{ В}$, де I_{K0} – тепловий струм, який в режимі відсікання транзистора тече з колектора в базу.

Вважаючи, що $\tau_{\alpha} \approx t_{\phi}$, визначаємо вимоги до транзистора за критерієм швидкодії:

$$f_{\alpha} \geq \frac{1}{2\pi t_{\phi}} = \frac{1}{2\pi \cdot 5 \cdot 10^{-7}} = \frac{10^6}{\pi} \approx 0.3 \text{ МГц}.$$

Однак на практиці цю нерівність необхідно суттєво підсилити. Враховуючи також критерій надійності, вибираємо транзистор КТ375А, призначений для роботи в імпульсних схемах ($U_{K. \text{нрпн}} = 60 \text{ В} \geq E_K = 18 \text{ В}$).

За залежністю з довідника $\beta(I_K)$ визначаємо область максимальних значень β : $I_K(\beta > \beta_{\min} = 20) = (5 \div 60) \text{ мА}$. Виходячи з того, що вимоги за швидкодією транзисторного ключа не жорсткі, з точки зору економічності режиму вибираємо $I_{K.H} = 10 \text{ мА}$. Тоді розраховуємо

$$R_K = \frac{E_K - U_{K.H}}{I_{K.H}} = \frac{18 - 0.4}{0.01} = 1760 \text{ Ом},$$

де $U_{K.H} = 0.4 \text{ В}$ – напруга насичення переходу колектор – емітер транзистора КТ375А. Вибираємо $R_K = 1.8 \text{ кОм}$.

За максимальним значенням зворотного струму колектора $I_{K0} = 10 \text{ мкА}$ при $t = +85^\circ \text{C}$ перевіримо, чи виконується умова $|U^1| \geq 9.5 \text{ В}$:

$$\begin{aligned} U^1 &= -(E_K - I_{K0}R_K) = -(18 - 10^3 \cdot 10 \cdot 10^{-6}) = \\ &= -17.982 \text{ В} < -9.5 \text{ В} \end{aligned}$$

Таким чином, вимога до рівня логічної одиниці вихідної напруги виконується.

Розглянемо роботу транзисторного ключа в двох характерних режимах – відсікання та насичення.

Режим відсікання. Використовуючи до подільника напруги R_1, R_2 теорему про еквівалентний генератор та замінюючи транзистор VT генератором струму I_{K0} , представимо еквівалентну схему вхідного кола транзисторного ключа (рис. 4.16, а). Для забезпечення режиму відсікання напруга на базі транзистора має задовольняти умову (рис. 4.16, б)

$$U_{\bar{o}}^+ \geq -|U_{\bar{o}0}| + U_3^0,$$

де $U_{\bar{o}0}$ – порогова напруга відкривання транзистора, $U_{\bar{o}0} = -0.6 \text{ В}$; U_3^0 – завадостійкість, тобто максимально допустимий рівень завади від’ємного знаку (в нашому випадку) при напрузі запирання на вході.

З еквівалентної схеми знаходимо

$$U_{\bar{o}}^+ = E^1 \frac{R}{R + R_{\bar{o}}} - E_{\bar{o}} \frac{R_{\bar{o}}}{R + R_{\bar{o}}} - I_{K0} \frac{R R_{\bar{o}}}{R + R_{\bar{o}}} \geq U_3^0 - |U_{\bar{o}0}|,$$

$$\text{де } E_{\bar{o}} = K_R E_K, \quad K_R = \frac{R_2}{R_1 + R_2}, \quad R = \frac{R_1 R_2}{R_1 + R_2}.$$

З останнього співвідношення

$$R_{\bar{o}} \leq R \frac{E^1 + |U_{\bar{o}0}| - U_3^0}{K_R E_K + I_{K0} R + U_3^0 - |U_{\bar{o}0}|}.$$

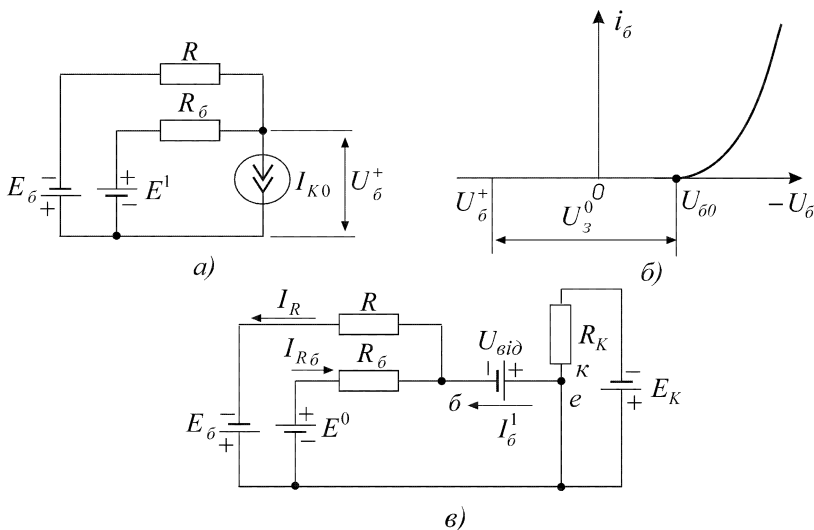


Рис. 4.16. Еквівалентна схема вхідного кола транзисторного ключа в режимі відсікання (а), вхідна характеристика транзистора (б) та еквівалентна схема транзисторного ключа в режимі насичення (в)

При $E^1 \gg |U_{\delta 0}| - U_3^0$ та $K_R E_K \gg I_{K0} R + U_3^0 - |U_{\delta 0}|$ (що виконується на практиці) отримаємо умову відсікання:

$$R_{\delta} \leq \frac{E^1 R}{K_R E_K}. \quad (4.6)$$

Якщо врахувати, що $\frac{R}{K_R} = \frac{R_1 R_2}{R_1 + R_2} \cdot \frac{R_1 + R_2}{R_2} = R_1$, то умову відсікання можна переписати таким чином:

$$R_{\delta} \leq R_1 \frac{E^1}{E_K}. \quad (4.7)$$

Крім того, з еквівалентної схеми видно, що через джерело сигналу E^1 протікає струм

$$I_{\delta x} \approx \frac{E^1 + K_R E_K}{R + R_{\delta}}.$$

Режим насичення. Представимо транзистор в режимі насичення в якості джерела $U_{\epsilon i \partial} = U_{\bar{o}.H}$ та отримаємо еквівалентну схему транзисторного ключа (рис. 4.16, в), з якої визначимо:

$$I_{\bar{o}}^1 = I_R - I_{R\bar{o}}; \quad I_R = \frac{E_{\bar{o}} - |U_{\epsilon i \partial}|}{R}; \quad I_{R\bar{o}} = \frac{|U_{\epsilon i \partial}| + E^0}{R_{\bar{o}}}.$$

Для вводу транзистора в режим насичення з потрібним коефіцієнтом S необхідно забезпечити струм

$$I_{\bar{o}}^1 = S I_{\bar{o}.H} = S \frac{I_{K.H}}{\beta} \approx \frac{S}{\beta} \cdot \frac{E_K}{R_K}.$$

В цьому випадку з останніх чотирьох співвідношень знайдемо умову насичення:

$$R = R_K \frac{\beta}{S} \left[K_R - \frac{|U_{\epsilon i \partial}|}{E_K} - \left(K_R + \frac{U_3^1}{E_K} \right) \frac{U_{\epsilon i \partial} + E^0}{E^1 - U_3^1} \right] \approx \quad (4.8)$$

$$\approx R_K \frac{K_R \beta}{S}.$$

Тоді

$$R_{\bar{o}} \leq R \frac{E^1}{K_R E_K} = R_K \frac{K_R \beta}{S} \frac{E^1}{K_R E_K} = R_K \frac{\beta}{S} \frac{E^1}{E_K},$$

при цьому потрібний коефіцієнт насичення $S \geq (1.2 \div 1.5)$ має забезпечуватися при мінімальному значенні β . Приймаємо $S = 1.4$ при $\beta_{min} = 20$ та розраховуємо значення $R_{\bar{o}}$:

$$R_{\bar{o}} \leq 1800 \cdot \frac{20}{1.4} \cdot \frac{2.4}{18} = 3.428 \text{ кОм}.$$

Вибираємо $R_{\bar{o}} = 3.3 \text{ кОм}$.

Для обмеження струму, споживаного від джерела сигналу, необхідно забезпечити виконання умови

$$R + R_{\bar{o}} \geq \frac{E^1 + K_R E_K}{I_{\epsilon x \max}}.$$

Підставивши в останнє співвідношення результати виразу (4.8) та провівши еквівалентні перетворення, для мінімального значення β_{min} отримаємо

$$K_R \geq \frac{S(E^1 - R_{\delta} I_{ex.max})}{R_K \beta_{min} I_{ex.max} - S E_K}. \quad (4.9)$$

За співвідношенням (4.9) можна визначити мінімальне значення коефіцієнта передачі резистивного подільника напруги $K_{R.min}$. Підставивши чисельні дані, отримаємо

$$K_{R.min} = \frac{1.4 \cdot (2.4 - 3300 \cdot 10^{-3})}{1800 \cdot 20 \cdot 10^{-3} - 1.4 \cdot 18} = -0.117.$$

Значення коефіцієнта передачі K_R необхідно обмежити з верхнього боку, тому для режиму відсікання буде справедливим таке співвідношення

$$E^1 - |U_{3.min}^0| \geq K_R |E_K| - |U_{\delta 0}|,$$

з якого

$$K_{R.max} = \frac{E^1 - |U_{3.min}^0| + |U_{\delta 0}|}{|E_K|}.$$

Підставивши чисельні значення, отримаємо

$$K_{R.max} = \frac{2.4 - 0.5 + 0.6}{18} = 0.139.$$

Однак реально значення $K_{R.max}$ можна дещо збільшити, тому що ми не врахували вплив подільника на вході ключа. Тому на основі проведених розрахунків вибираємо $K_R = 0.3$.

Далі визначимо опір резистора верхнього плеча подільника напруги за формулою

$$R_1 \geq R_{\delta} \frac{E_K}{E^1} = 3300 \cdot \frac{18}{2.4} = 24.75 \text{ кОм}.$$

Для подальших розрахунків вибираємо $R_1 = 27 \text{ кОм}$.

$$R_2 = \frac{K_R R_1}{1 - K_R} = \frac{0.3 \cdot 27}{1 - 0.3} = 11.57 \text{ кОм}.$$

Для подальших розрахунків вибираємо $R_2 = 12 \text{ кОм}$.

Для знайдених номіналів резисторів R_1 і R_2 обчислимо:

$$K_R = \frac{R_2}{R_1 + R_2} = \frac{12}{27 + 12} = 0.308;$$

$$R = \frac{R_1 R_2}{R_1 + R_2} = \frac{27 \cdot 12}{27 + 12} = 8.31 \text{ кОм}.$$

3. Розраховуємо потрібні для побудови часових діаграм величини.

$$\begin{aligned} U_{\bar{o}}(t_1) &= U_{\bar{o}}^+ = E^1 \frac{R}{R + R_{\bar{o}}} - E_{\bar{o}} \frac{R_{\bar{o}}}{R + R_{\bar{o}}} - I_{K0} \frac{R R_{\bar{o}}}{R + R_{\bar{o}}} = \\ &= 2.4 \frac{8310}{8310 + 3300} - 0.308 \cdot 18 \frac{3300}{8310 + 3300} - 10^{-5} \frac{8310 \cdot 3300}{8310 + 3300} = \\ &= 1.718 - 1.576 - 0.024 = 0.118 \text{ В}. \end{aligned}$$

Після моменту t_1 напруга на вході зменшується до величини $E = E^0 = 0.4 \text{ В}$, паразитна ємність C_{BX} намагається перезарядитися до напруги $U_{\bar{o}}(\infty)$, яку знайдемо за виразом

$$\begin{aligned} U_{\bar{o}}(\infty) &= E^0 \frac{R}{R + R_{\bar{o}}} - K_R E_K \frac{R_{\bar{o}}}{R + R_{\bar{o}}} - I_{K0} \frac{R R_{\bar{o}}}{R + R_{\bar{o}}} = \\ &= 0.4 \frac{8310}{8310 + 3300} - 0.308 \cdot 18 \frac{3300}{8310 + 3300} - 10^{-5} \frac{8310 \cdot 3300}{8310 + 3300} = \\ &= 0.286 - 1.576 - 0.024 = -1.314 \text{ В}. \end{aligned}$$

Таким чином, в розглянутому транзисторному ключі реалізується необхідна за завданням завадостійкість, тому що при дії на вході транзисторного ключа сигналу рівня логічної одиниці негативна завада з амплітудою $|U_{3,min}| = 0.5 \text{ В}$ не буде приводити до відкриття транзистора, тому що

$U_{\bar{\sigma}}(t_1) - |U_{3.min}| = -0.382 \text{ В} > U_{\bar{\sigma}0} = -0.6 \text{ В}$. Аналогічно, при дії на вході транзисторного ключа сигналу рівня логічного нуля позитивна завада з амплітудою $|U_{3.min}| = 0.5 \text{ В}$ не буде приводити до виходу транзистора з режиму насичення, тому що $U_{\bar{\sigma}}(\infty) + |U_{3.min}| = -0.814 \text{ В} < U_{\bar{\sigma}id} = -0.8 \text{ В}$.

Затримка включення або час підготовки транзистора до відкривання

$$t_3^{10} \approx \tau \ln \left[1 + \left| \frac{U_{\bar{\sigma}}^+}{U_{\bar{\sigma}}(\infty)} \right| \right] = \tau \ln \frac{E^1 - E^0}{E^1 - |U_{\bar{\sigma}0}|},$$

де $C_{BX} = C_K + C_E + C_H = (80 \dots 120) \text{ нФ}$ – вхідна паразитна ємність транзисторного ключа; $\tau = C_{BX} \frac{R \cdot R_{\bar{\sigma}}}{R + R_{\bar{\sigma}}}$ – стала часу вхідного кола транзисторного ключа.

Підставивши чисельні дані, отримаємо

$$t_3^{10} = 100 \cdot 10^{-12} \cdot \frac{8310 \cdot 3300}{8310 + 3300} \ln \left[1 + \left| -\frac{0.118}{1.314} \right| \right] = 0.02 \text{ мкс}.$$

Знайдемо струм бази

$$\begin{aligned} I_{\bar{\sigma}}^1 &= I_R - I_{R\bar{\sigma}} = \frac{E_{\bar{\sigma}} - |U_{\bar{\sigma}id}|}{R} - \frac{|U_{\bar{\sigma}id}| + E^0}{R_{\bar{\sigma}}} = \frac{K_R E_K - |U_{\bar{\sigma}id}|}{R} - \\ &- \frac{|U_{\bar{\sigma}id}| + E^0}{R_{\bar{\sigma}}} = \frac{0.308 \cdot 18 - 0.8}{8310} - \frac{0.8 + 0.4}{3300} = \\ &= 0.57 \cdot 10^{-3} - 0.36 \cdot 10^{-3} = 0.21 \text{ мА}, \end{aligned}$$

де $|U_{\bar{\sigma}id}| = |U_{\bar{\sigma}e \text{ нас}}| = 0.8 \text{ В}$ – напруга насичення переходу база – емітер транзистора.

Для значення коефіцієнта насичення $S = 1.4$ обчислимо:

$$I_{\bar{\sigma}.H} = \frac{I_{\bar{\sigma}}^1}{S} = \frac{0.21}{1.4} = 0.15 \text{ мА};$$

$$\beta = \frac{I_{K.H}}{I_{\bar{o}.H}} = \frac{10}{0,15} = 66.7 < \beta_{max} = 100;$$

$$I_K^1 = S I_{K.H} = 1.4 \cdot 10 = 14 \text{ мА};$$

$$\tau_\beta = \beta \tau_\alpha = \frac{\beta}{2\pi f_\alpha} = \frac{66.7}{2\pi \cdot 150 \cdot 10^6} = 0.07 \text{ мкс};$$

$$\tau_{\beta_{екв}} = \tau_\beta + \beta C_K R_K = 0.07 \cdot 10^{-6} +$$

$$+ 66.7 \cdot 2.7 \cdot 10^{-12} \cdot 1800 = 0.394 \text{ мкс};$$

$$t_\phi^{10} = \tau_{\beta_{екв}} \ln \frac{1}{1 - \frac{1}{S}} = \tau_{\beta_{екв}} \ln \frac{S}{S-1} =$$

$$= 0.394 \cdot 10^{-6} \ln \frac{1.4}{1.4-1} = 0.493 \text{ мкс} < 0.5 \text{ мкс};$$

$$\tau_H \approx \tau_\beta \Rightarrow t_H = 3\tau_H = 3 \cdot 0.07 \cdot 10^{-6} = 0.21 \text{ мкс}.$$

4. За розрахованими даними будемо часові діаграми, синхронні з $E(t)$, наведені на рис. 4.17.

5. З'ясуємо, як зміняться величини, розраховані в п.3, при збільшенні рівня E^1 та незмінних інших параметрах транзисторного ключа. Позначимо для компактності збільшення (\uparrow) та зменшення (\downarrow), тоді отримаємо:

$$E^1 \uparrow \Rightarrow U_{\bar{o}}^+ \uparrow; E^1 \uparrow \Rightarrow t_3^{10} \uparrow.$$

Якщо всі інші параметри схеми транзисторного ключа незмінні, то на тривалість фронту t_ϕ^{10} зміна E^1 не впливає. Таким чином, збільшення E^1 приводить до збільшення часу затримки включення, оскільки паразитний конденсатор C_{BX} , заряджений до $U_{\bar{o}}^+$, буде довше перезаряджатися до порогового значення напруги $-|U_{\bar{o}0}|$, однак в нашому випадку це несуттєво, тому що $t_3^{10} = 0.02 \text{ мкс} \ll t_\phi^{10} = 0.493 \text{ мкс}$.

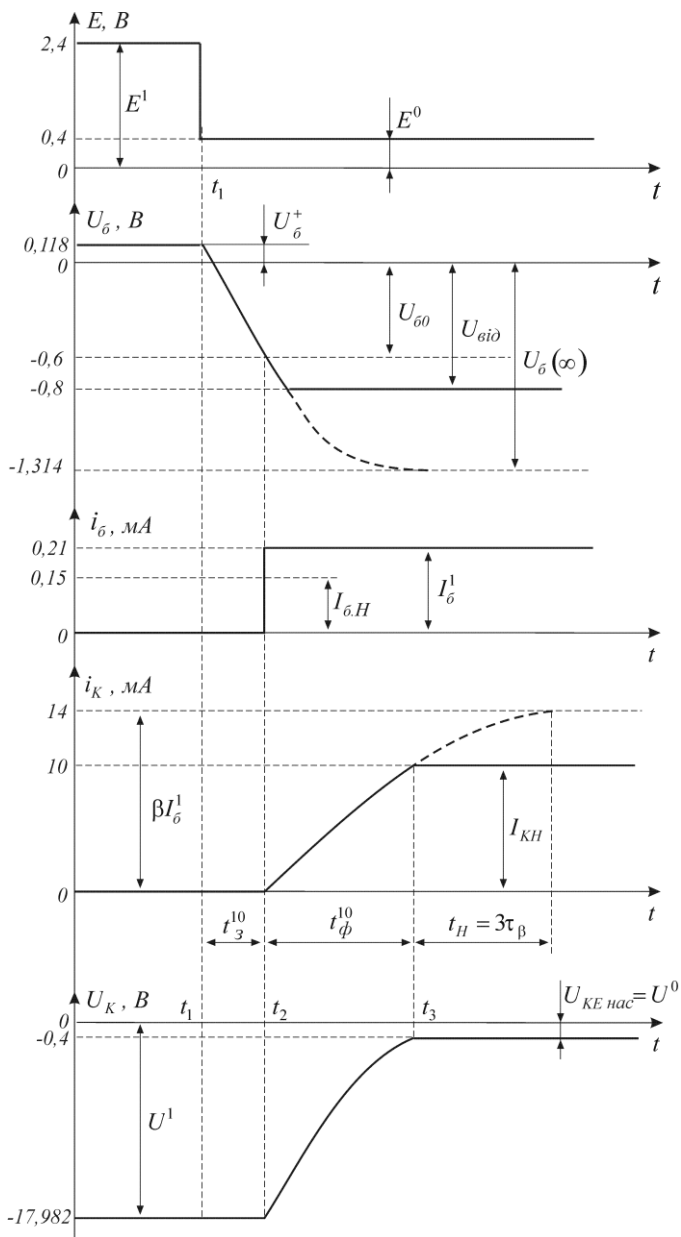


Рис. 4.17. Часові діаграми роботи транзисторного ключа

Контрольні питання

1. Поясніть роботу ідеального ключа.
2. Наведіть недоліки діодних ключів?
3. Які особливості роботи транзистора в ключовому режимі?
4. Яка роль резистора R_B у вхідному колі транзисторного ключа?
5. Як транзисторні ключі змінюють фазу сигналів?
6. Які причини спотворення форми імпульсів у транзисторних ключах?
7. Назвіть приклади застосування електронних ключів?
8. Які фактори впливають на швидкодію транзисторних ключів?
9. Наведіть повні і спрощені лінійні моделі транзисторів в областях насичення і відсікання. Чому дорівнюють коефіцієнти передачі транзистора в цих областях?
10. Наведіть методику розрахунку транзисторного ключа в режимі насичення, модель ключа, а також поясніть зміст коефіцієнта насичення S і його рекомендовані значення.
11. Викладіть методику розрахунку транзисторного ключа в режимі відсікання, а також наведіть модель ключа в даному режимі.

5. АРИФМЕТИЧНІ ОСНОВИ ЦИФРОВОЇ ТЕХНІКИ

5.1. Перетворення чисел між системами числення

5.1.1. Системи числення

У *позиційній* системі числення (СЧ), на відміну від *непозиційної*, наприклад, римської, кількість, позначувана цифрою x_i , залежить від її позиції – розряду i в числі

$$X_C = x_{n-1}x_{n-2} \dots x_1x_0, x_{-1}x_{-2} \dots x_{-m}, \quad (5.1)$$

де n, m – кількість цифр відповідно в цілій та дробовій частинах числа, відокремлюваних одна від однієї розділовою комою або крапкою. Найбільшого поширення набули позиційні СЧ, в яких вага розрядів c^i (кількісне значення кожної їх одиниці) змінюється в геометричній прогресії, знаменник якої c є *основною* СЧ, що дорівнює кількості різних цифр, потрібних для запису довільного числа. Отже, запис (5.1) відповідає кількості

$$X_c = x_{n-1}c^{n-1} + \dots + x_0c^0 + \dots + x_{-m}c^{-m} = \sum_{i=n-1}^{-m} x_i c^i, \quad (5.2)$$

наприклад, $904,1_{10} = 9 \cdot 10^2 + 0 \cdot 10^1 + 4 \cdot 10^0 + 1 \cdot 10^{-1}$.

У найуживаніших СЧ (табл. 5.1) використовуються арабські цифри, а якщо їх бракує – також літери латинської абетки: цифри A, \dots, F відповідають десятковим числам $10, \dots, 15$. (Цифри можуть позначатися і від'ємними числами, наприклад, у знакорозрядних формах двійкової і трійкової СЧ застосовуються цифри $1, 0, 1$, де $1 = -1$). Крім індексних використовуються також літерні позначення СЧ: D (Decimal), B (Binary), H (Hexadecimal), Q (Octal), наприклад, $94,1 = 94,1_{10} = 94,1D$; $10,01_2 = 10,01B$; $7B, A_{16} = 7B, AH$; $76,5_8 = 76,5Q$.

У табл. 5.2 наведено приклади на деякі числа в основних СЧ. Якщо числа в різних СЧ відповідають однаковій кількості, вони є еквівалентними, наприклад, $10101100,1_2 = 254,4_8 =$

$= 172,5_{10} = AC,8_{16}$; найдовшими є двійкові числа, найкоротшими – шістнадцяткові, а для людини, звичайно, зручнішими – десяткові.

Таблиця 5.1. Символи, що використовуються в СЧ

Основа	10	2	8	16
Найменування Цифри	десятькова 0,1,...,9	двійкова 0,1	вісімкова 0,1,...,7	шістнадцяткова 0,1,...,9,A,B,...,F

Як відомо, у цифровій техніці використовується переважно двійкова СЧ, бо їй властиві такі позитивні якості: найпростіша реалізація елемента для відображення всього двох цифр; найбільша завадостійкість, зумовлена тим, що під час дії завад, дестабілізівних чинників тощо легше розрізнити два рівні напруги, ніж кілька; простота виконання арифметичних та логічних операцій і пов'язана з цим висока швидкодія; ощадливість та можливість уніфікації обладнання.

5.1.2. Перетворення між двійковою, шістнадцятковою та вісімковою системами числення

Недолік двійкової СЧ (незручність запису довгих чисел) компенсується широким запровадженням шістнадцяткової та вісімкової систем для сполучення між пристроями цифрової техніки. Дані системи числення вибрано тому, що між ними й двійковою системою найлегше виконати перетворення чисел.

Взагалі, якщо основи двох систем співвідносяться як $c_2 = c_1^k$ (k – ціле число), то для перетворення $X_{C_1} \rightarrow X_{C_2}$ (чисел від системи з основою c_1 до системи з основою c_2) досить розбити вихідне число X_{C_1} на групи з k цифр відносно коми та замінити кожен групу на цифру системи з основою c_2 . Так, для перетворень $X_2 \rightarrow X_{16}$, $X_2 \rightarrow X_8$ (прикладі на рис. 5.1) двійкове число розбиваємо відносно коми ліворуч та праворуч (зазначено стрілками вгору) відповідно по чотири (тетради) і три (тріади) розряди, додаємо в крайніх групах, за необхідністю, незначущі нулі та замінюємо утворені тетради й тріади симво-

Таблица 5.2. Представления чисел в основных СЧ

10	2	8	16	10	2	8	16	10	2	8	16
0	0	0	0	16	10000	20	10	1/2	0,1	0,4	0,8
1	1	1	1	17	10001	21	11				
2	10	2	2	18	10010	22	12	1/4	0,01	0,2	0,4
3	11	3	3	19	10011	23	13				
4	100	4	4	20	10100	24	14	1/8	0,001	0,1	0,2
5	101	5	5	21	10101	25	15				
6	110	6	6	22	10110	26	16	1/16		0,04	0,1
7	111	7	7	23	10111	27	17				
8	1000	10	8	24	11000	30	18	1/64		0,01	0,04
9	1001	11	9	25	11001	31	19				
10	1010	12	<i>A</i>	26	11010	32	<i>1A</i>	1/256		0,002	0,01
11	1011	13	<i>B</i>	27	11011	33	<i>1B</i>				
12	1100	14	<i>C</i>	28	11100	34	<i>1C</i>	1/512		0,001	0,008
13	1101	15	<i>D</i>	29	11101	35	<i>1D</i>				
14	1110	16	<i>E</i>	30	11110	36	<i>1E</i>	1/4096			0,001
15	1111	17	<i>F</i>	31	11111	37	<i>1F</i>				

Рис. 5.1. Приклади перетворень типу $X_2 \leftrightarrow X_{16}$; $X_2 \leftrightarrow X_8$

Часто доводиться перетворювати числа з будь-якої СЧ до звичної десяткової системи. Універсальним для цього є метод за *формулою полінома* (5.2). Для СЧ з великими основами, наприклад, $c=16$ (рис. 5.2, *a*), можна скористатися таблицями

степенів та калькулятором, а для двійкової системи досить підсумувати вагові коефіцієнти (на рис. 5.2, б розташовані вгорі), що відповідають одиничним розрядам двійкового числа.

Якщо у виразі (5.2) поступово виносити за дужки основу в першому степені, одержимо модифіковану формулу, в якій менше операцій множення та яку легше запрограмувати. Обчислення за нею (*алгоритм Горнера*) розглянемо на прикладі (рис. 5.2, в). Старший розряд числа (B) множимо на основу системи ($c=16$), до добутку додаємо наступний розряд (A) і суму (186) знов множимо на основу; продовжуючи цей ланцюжок, під кінець до попереднього добутку додаємо останній розряд (0) цілої частини вихідного числа й дістаємо, таким чином, цілу частину десяткового числа (2976). Перетворення дробової частини виконується аналогічно, але в протилежному напрямку. Молодший розряд дробової частини (F) ділимо на основу системи (або множимо на $1/c=0,0625$), до результату додаємо наступний ліворуч розряд (8) і суму (8,9375) знов ділимо на основу. Після додавання останнього (першого після коми) розряду (0) також виконуємо ділення і дістаємо дробову частину числа (0,0349...). У підсумку маємо $BA0,08F_{16} \approx 2976,0349_{10}$.

Перетворення до десяткової СЧ за алгоритмом Горнера можна робити і з двійкової системи, проте для довгих чисел воно коротше за схемою $X_2 \rightarrow X_{16} \rightarrow X_{10}$ (рис. 5.2, г).

5.1.4. Перетворення від десяткової до інших систем числення

Для зображення чисел та іншої інформації в цифрових пристроях часто необхідно перетворювати десяткові числа до СЧ з будь-якою основою c . *Універсальний алгоритм* такого перетворення розглянемо на прикладі $X_{10} \rightarrow X_{16}$ (рис. 5.3, а).

Цілу частину вихідного числа (2976) і кожну наступну частку (186) ділять послідовно на основу нової системи ($c=16$) до отримання неподільної частки (11). Записуючи останню частку (11) і залишки від ділення (10; 0) як цифри нової системи (B ; A ; 0) у зворотному порядку їх здобуття, дістають цілу частину нового числа ($BA0_{16}$).

$$X_c \rightarrow X_{10}$$

$$BA0,08F_{16} = 11 \cdot 16^2 + 10 \cdot 16^1 + 0 \cdot 16^0 + 0 \cdot 16^{-1} + 8 \cdot 16^{-2} + 15 \cdot 16^{-3} \approx 2976,035_{10}$$

a

$$\begin{array}{cccccccc} 16 & 8 & 4 & 2 & 1 & 1/2 & 1/4 & 1/8 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1_2 \\ \downarrow & & & \downarrow & \downarrow & & \downarrow & \downarrow \\ 16 & + & & 2 + 1 & + & 0,25 + 0,125 & \longrightarrow & 19,375_{10} \end{array}$$

б

$$\begin{array}{r} \begin{array}{l} \times \quad 11 \leftarrow \\ \hline 16 \\ + \quad 10 \leftarrow \\ \hline 186 \\ \times \quad 16 \\ + \quad 0 \leftarrow \\ \hline 2976 \end{array} \quad \begin{array}{l} BA0,08F_{16} \\ \downarrow \\ 2976 \end{array} \quad \begin{array}{l} \begin{array}{l} \rightarrow 15 \mid 16 \\ + 0,9375 \\ \hline \rightarrow 8 \\ 8,9375 \mid 16 \\ + 0,5586 \\ \hline \rightarrow 0 \\ 0,5586 \mid 16 \\ \hline 0,0349 \end{array} \\ \downarrow \\ 0,0349_{10} \end{array} \\ \hline 2976,0349_{10} \end{array}$$

в

$$\begin{array}{l} 0001\ 0011, 0110_2 = 13,6_{16} = 19,375_{10} \\ \begin{array}{l} \times \quad 1 \leftarrow \\ \hline 16 \\ + \quad 3 \leftarrow \\ \hline 19 \end{array} \quad \begin{array}{l} \rightarrow 6 \mid 16 \\ \hline 0,375 \end{array} \end{array}$$

г

Рис. 5.2. Приклади перетворень типу $X_C \rightarrow X_{10}$

Дробову частину вихідного числа (0,035) і дробову частину кожного наступного добутку (0,56; 0,96; ...) тепер множать послідовно на ту ж саму основу (16) до отримання нуля в дробовій частині чергового добутку або потрібної точності (кількості розрядів після коми). Цілі частини добутків від кожного множення (0; 8; 15; ...), записані як цифри нової СЧ (0; 8; F) у прямому порядку їх одержання, складають дробову частину нового числа (0,08F...₁₆). Таким чином, $2976,035_{10} \approx BA0,08F_{16}$.

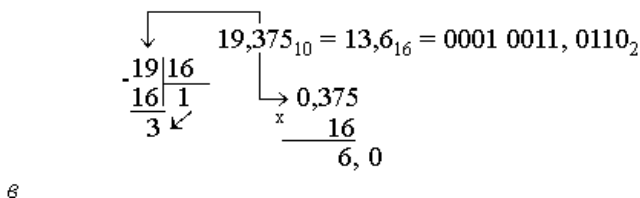
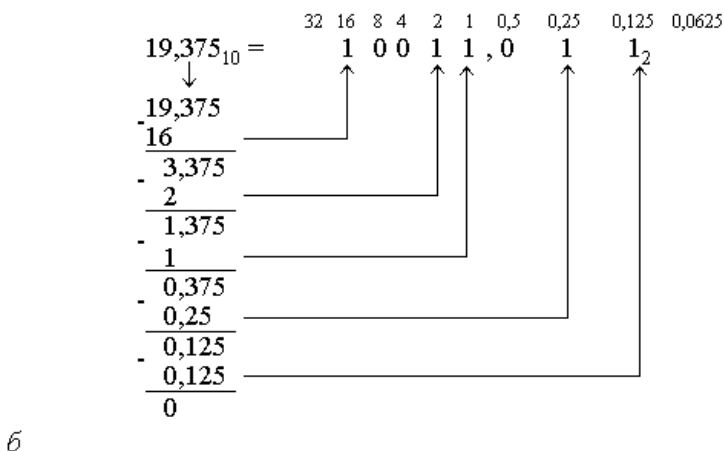
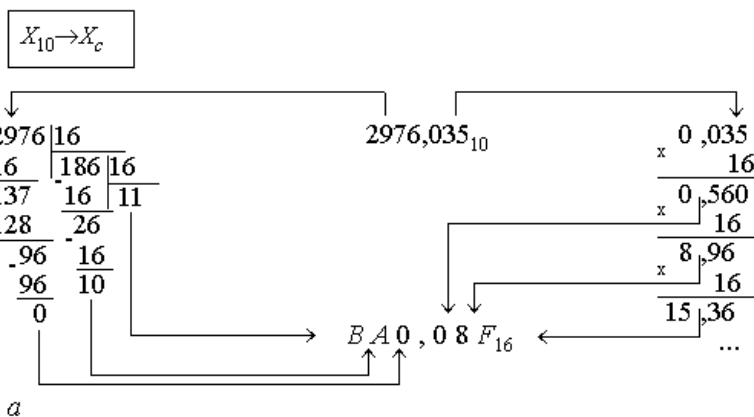


Рис. 5.3. Приклади перетворень типу $X_{10} \rightarrow X_C$

Цей алгоритм застосовується і для перетворення $X_{10} \rightarrow X_2$, а з огляду на те, що двійкові вагові коефіцієнти дуже легко визначаються, це перетворення можна виконати послідов-

ним відніманням (рис. 5.3, б) від десяткового числа $(19,375)$ і кожної наступної різниці $(3,375; 1,375; \dots)$ найбільших вагових коефіцієнтів, що не перевищують зменшуваного $(16 < 19,375; 2 < 3,375; \dots)$. Лишається тільки записати одиниці до тих розрядів двійкового числа, що відповідають від'ємникам, а до решти розрядів – нулі. Кількість операцій у порівнянні з попереднім алгоритмом тут зменшується на кількість нулів у двійковому числі (наприклад, у перетворенні $1025_{10} \rightarrow X_2$ зменшується на порядок). Проте великі десяткові числа швидше перетворити в двійкові за схемою $X_{10} \rightarrow X_{16} \rightarrow X_2$ (рис. 5.3, в).

5.1.5. Деякі спрощення перетворень між системами числення

Звернімо увагу на те, що для перетворень між будь-якою системою X_c та десятковою X_{10} ми користувалися алгоритмами, в яких арифметичні дії виконуються в десятковій системі. Якщо необхідно перетворювати числа між довільними системами $X_{c1} \rightarrow X_{c2}$, крім випадку $c_2 = c_1^k$, то операції доводиться здійснювати в арифметиці довільної системи, що викликає незручності. Тому варто йти простим шляхом $X_{c1} \rightarrow X_{10} \rightarrow X_{c2}$, коли всі дії можна виконувати десятковою арифметикою, як, наприклад, зроблено перетворення $X_3 \rightarrow X_5$ (рис. 5.4). В алгоритмах машинного перетворення, навпаки, зручніше виконувати арифметичні дії у двійковій системі, наприклад, шляхом послідовного ділення і множення на число 1010_2 .

Враховуючи, що зсув розряду числа ліворуч або праворуч, еквівалентний відповідно множенню або діленню на основу системи, можна спростити деякі перетворення, користуючись співвідношеннями, наприклад, для двійкової системи (три крапки означають, що цифра повторюється в k розрядах поспіль):

$1\dots1 = 2^k - 1$; $0,1\dots1 = 1 - 2^{-k}$; $X0\dots0 = X \cdot 2^k$; $0,0\dots0X = X \cdot 2^{-(k+n)}$, де n – кількість розрядів числа X .

$$X_{c1} \rightarrow X_{c2}$$

$$201,1_3 = 2 \cdot 3^2 + 1 + 1/3 = 19,(3)_{10} = 34,(13)_5$$

$$\begin{array}{r} \downarrow \qquad \qquad \downarrow \\ \begin{array}{r} 19 \mid 5 \\ 15 \mid 3 \\ \hline 4 \end{array} \quad \begin{array}{r} 0,(3) \\ \times 5 \\ \hline 1,(6) \\ \times 5 \\ \hline 3,(3) \end{array} \end{array}$$

Рис. 5.4. Приклад перетворення типу $X_{C1} \rightarrow X_{C2}$

Приклади. 1) $11111,1111_2 = (2^5 - 1) + (1 - 2^{-4}) = 31 + 15/16$;

2) $111100101,111100101_2 = (2^4 - 1) \cdot 2^5 + 5 + (1 - 2^{-4} + 5 \cdot 2^{-9}) =$
 $= 15 \cdot 2^5 + 5 + 1 - (2^5 - 5) \cdot 2^{-9} = 485 + (1 - 27/512) = 485 + 485/512.$

Як і слід було очікувати, дробову частину можна перетворити як ціле число, а відтак помножити його на 2^{-m} , де m – кількість розрядів після коми. Природно, аналогічно спрощуються перетворення і з інших систем до десяткової. Наприклад, $1400.FF_{16} = 20 \cdot 16^2 + (1 - 16^{-2}) = 5120 + 255/256.$

5.2. Інформаційна ємність та форми зображення двійкових чисел

5.2.1. Одиниці кількості цифрової інформації

Змінюючи всі можливі сполучення цифр x_i в (5.2), одержимо (вважаючи для простоти $m=0$) набір цілих чисел, які можна зобразити n розрядами $X_c = 0 \dots (c^n - 1)$, тобто разом $K = c^n$ різних чисел. Наприклад, 3 розряди десяткової системи утворюють 10^3 чисел $X_{10} = 0, \dots, 999$. Так само діапазон зображуваних n розрядами двійкових чисел становить $X_2 = 0, \dots, (2^n - 1)$, а їх кількість $K = 2^n$ удвічі більша за вагу старшого розряду 2^{n-1} .

Чим більша розрядність цифрового пристрою n , тим менша ймовірність $p(X_i) = 1/K$ (для рівноймовірних подій) прий-

няття ним від джерела деякого конкретного повідомлення – одного з діапазону K зображуваних чисел – і тим більша кількість інформації (як міра невизначеності повідомлення) у ньому міститься. У теорії інформації прийнята логарифмічна міра її кількості, що для рівномірних подій становить

$$\log_2 = \frac{1}{p(X_i)} = \log_2 K = n,$$

тобто визначається кількістю двійкових розрядів, яку вміщує той чи інший пристрій.

Одиниця кількості двійкової інформації – біт (bit – binary digit – двійковий розряд; якщо використовуються десяткові логарифми, одиницею інформації є діт, якщо натуральні – ніт) тлумачиться як така, що міститься в одному двійковому розряді, або як змінна x_i , що може набувати тільки двох значень: 0 та 1; 1 біт інформації можна відтворювати транзистором у ключовому режимі, передавати однодротовою лінією, зберігати в одному елементі пам'яті (тригері та ін.). Зростання інформаційної ємності (табл. 5.3) потребує, природно, і ускладнення обладнання. Так, для ємності 1 байт = 8 біт вже потрібна комірка пам'яті з восьми однібітових елементів і восьмидротова лінія, яка називається шиною даних.

Таблиця 5.3. Визначення інформаційної ємності

Кількість двійкових розрядів, n	Діапазон даних, $D_2 (D_{10})$	Кількість чисел, K	Інформаційна ємність
1	0; 1 (0; 1)	2	1 біт
4	0000 ... 1111 (0...15)	16	1/2байта
8	00000000 ... 11111111(0...255)	256	1 байт
16	0...0 ... 1...1 (0...65535)	65536	2 байта

5.2.2. Машинне слово

Ємність оброблюваної інформації визначається для кожної ЕОМ розрядністю цифрових пристроїв у її складі. Аби

підкреслити, що деяка ЕОМ за одним тактом обробляє двійкове число певної довжини, його називають *машинным словом*; у мікропроцесорній техніці частіше використовуються слова довжиною 1 байт (рис. 5.5, *а*) і 2 байти. Можуть застосовуватися також слова половинної (півслово) або подвійної довжини. Якщо слова зберігаються в комітках пам'яті ЕОМ, то для звернення до них (наприклад, для зчитування) необхідно зазначити їх адреси – нумерацію комірок, де вони розташовані (рис. 5.5, *б*). Для адресування теж використовується двійкова система, проте занотовувати та вводити адреси зручніше в шістнадцятковій системі (табл. 5.4). Якщо в комірці пам'яті зберігається слово в 1 байт інформації, то за допомогою одного двійкового розряду адреси можна звернутися лише до двох комірок (одна з них активізується нулем, друга – одиницею), тобто до 2 байт інформації, за допомогою 10 розрядів – до 1024 байт = 1 Кбайт, а 20 розрядів – до 1024 Кбайт = 1 Мбайт.

Таблиця 5.4. Адресування до пам'яті

Кількість двійкових розрядів, n	Діапазон адрес, A_{16} (A_{10})	Кількість адрес	Ємність адресованої пам'яті
1	0; 1 (0; 1)	2	2 байти
4	0...F(0...15)	16	16 байт
8	00...FF (0...255)	256	1/4байт
10	000...3FF (0...1 023)	$2^{10}=1024$	1 Кбайт
12	000...FFF (0...4 095)	$4 \cdot 2^{10}$	4 Кбайти
16	0000...FFFF(0...65 535)	$64 \cdot 2^{10}$	64 Кбайт
18	00000...3FFFF (0...262 143)	$256 \cdot 2^{10}$	1/4байта
20	00000...FFFFF (0...1 048 575)	2^{20}	1 Мбайт

У цьому розумінні йдеться про ємність адресованої пам'яті. Для адресування широко використовуються 16- та 20-розрядні адреси (рис. 5.5, *в*), що дозволяють звертатися до пам'яті ємністю відповідно 64 Кбайт та 1 Мбайт. За допомогою наведеної таблиці легше розподілити пам'ять між окремими її

блоками, наприклад, блок в 1 Кбайт з початковою адресою $A700H$ закінчується адресою $A700+3FF=AAFFH$ (див. рис. 5.5, б).

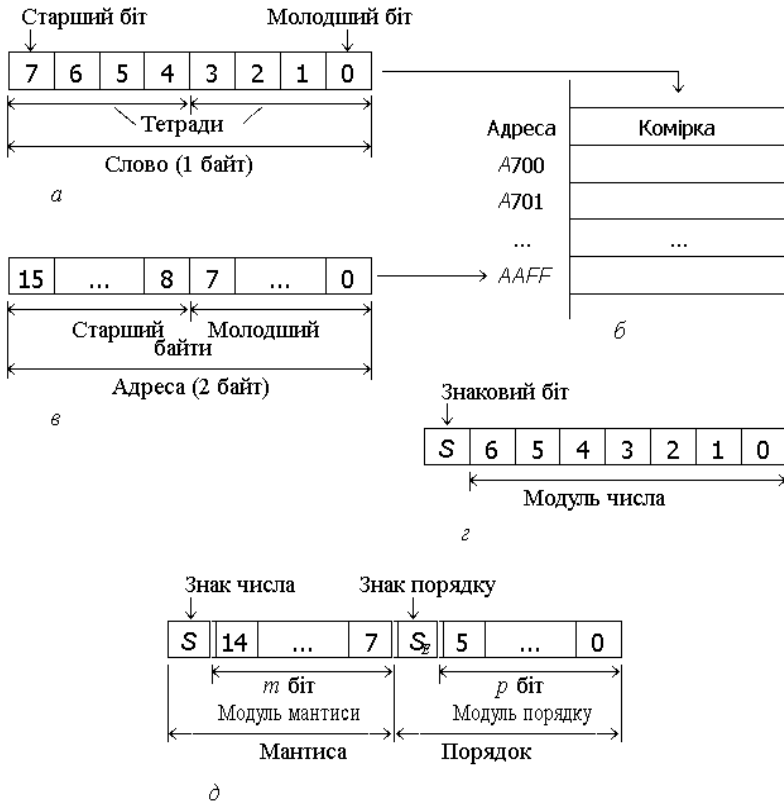


Рис. 5.5. До визначення адрес та форм зображення чисел

5.2.3. Форма зображення чисел з нерухомою комою

У форматі цілого числа зі знаком (рис. 5.5, г) один біт S (Sign – знак) призначається для знака: $S=0$ – число додатне, $S=1$ – від'ємне. Тому, наприклад, в однобайтовому слові на цифрову частину (модуль числа) залишається 7 біт, що становить діапазон зображуваних чисел від $1.1111111_2 = -127_{10}$ до $0.1111111_2 = +127_{10}$ (крапкою умовно відокремлено знаковий біт) або за мо-

дулем $0...127_{10}$.

Дробове число подається в такому ж форматі (див. рис. 5.5, з), проте кома фіксується зазвичай після знакового біта, тобто все число довжиною n біт сприймається як дріб: знаковий біт розташований у цілій частині, а решту $m=n-1$ розрядів дробової частини займає модуль числа. Отже, дані мають вписуватися до розрядної сітки шляхом добору масштабних коефіцієнтів. Таке подання чисел називають зображенням з *нерухомою* (фіксованою) *комою*. Найбільше число при цьому становить $X_{max}=0.1...1=1-2^{-m}$ (як і в десятковій системі, наприклад, $0,999=1-10^{-3}$), а найменше за модулем зображуване число, відмінне від нуля, $X_{min}=0,0...01=2^{-m}$, тобто діапазон зображуваних чисел за модулем становить $X_{min}..X_{max}=2^{-m}...1-2^{-m}$.

Під час зображення малих чисел старші розряди витискаються нулями і коли молодші розряди не вміщуються в розрядній сітці, вони не сприймаються ЕОМ (тому число, менше за 2^{-m} , називають *машинним нулем*), тому виникає абсолютна похибка, що не перевищує одиниці молодшого розряду $\Delta=2^{-m}$. Звідси дістаємо відносну похибку в усьому діапазоні чисел:

$$\delta \leq \frac{\Delta}{X_{min}} \dots \frac{\Delta}{X_{max}} = 1 \dots \frac{1}{2^m - 1}.$$

Наприклад, для однобайтового слова ($n=8, m=7$) маємо в діапазоні чисел $1/128...127/128$ відносну похибку біля 100% ... 0,8%, тобто, як і слід було очікувати, малі числа зображаються з дуже великою похибкою.

5.2.4. Форма зображення чисел з рухомою комою

У формі зображення з *рухомою* (пливною) *комою* (рис. 5.5, д) число подається в показниковому (або півлогарифмічному): половина зображення показникова) вигляді:

$$X=M \cdot 2^E,$$

де M – мантиса, E – порядок числа; два біти приділено знакам: S

– знаковий біт числа, S_E – знаковий біт порядку.

Здебільшого, модуль мантиси сприймається як правильний дріб (уявна кома є між бітом S та старшим бітом мантиси), а модуль порядку – завжди як ціле число.

Наприклад, якщо в двобайтовому слові 2 біти приділено знакам, 8 біт – модулю мантиси та 6 біт – модулю порядку, то зображення 1.01010000.0.000100 (розділові крапки між частинами слова зазначено умовно) є числом $X=(-0,0101 \cdot 2^{+100})_2 = -101_2 = -5_{10}$. Таке зображення вважається ненормалізованим, проте частіше використовується нормалізоване, до якого переходять за допомогою перації *нормалізації*: мантиса зсувається ліворуч доти, поки з'явиться одиниця після знакового біта числа (кожний крок зсуву еквівалентний множенню на 2), а порядок числа, аби воно не змінилося, зменшується на кількість кроків зсуву (еквівалентно діленню по черзі на 2). У нашому прикладі один крок ліворуч дає нормалізоване зображення числа

$$X=(-0,101 \cdot 2^{+11})_2 = -101_2 = -5_{10}.$$

У нормалізованому числі найменше значення мантиси, відмінне від нуля, становить $M_{\min}=0,10...0_2=2^{-1}$, її найбільша величина $M_{\max}=0,11...1_2=1-2^{-m} \approx 1$ та найбільше значення модуля порядку $E_{\max}=2^p-1$, де m, p – кількість розрядів у модулях відповідно мантиси та порядку. Звідси маємо модулі найменшого й найбільшого чисел:

$$X_{\min} = M_{\min} \cdot 2^{-E_{\max}} = 2^{-1} \cdot 2^{-(2^p-1)} = 2^{-2^p};$$

$$X_{\max} = M_{\max} \cdot 2^{+E_{\max}} = (1 - 2^{-m}) \cdot 2^{2^p-1} \approx 2^{2^p-1}$$

та діапазон їх зображення $X=2^{-2^p} \dots 2^{2^p-1}$, який у нашому прикладі при $m=8, p=6$ становить $X=2^{-64} \dots 2^{63}$.

Абсолютна похибка зображення мантиси, яка не перевищує 2^{-m} , визначає відносну похибку зображення чисел у всьому їх діапазоні (тут порядок числа не враховуємо, бо він входить до чисельників та знаменників):

$$\delta \leq \frac{\Delta}{M_{\min}} \dots \frac{\Delta}{M_{\max}} = \frac{1}{2^{m-1}} \dots \frac{1}{2^m - 1},$$

тобто в нашому прикладі змінюється в діапазоні $2^{-7} \dots 2^{-8} < 0,8\% \dots 0,4\%$.

З метою розширення діапазону зображуваних чисел без збільшення довжини розрядної сітки в сучасних ЕОМ числа з рухомою комою подаються в шістнадцятковій системі числення

$$X = M \cdot 16^E$$

з довжиною слова 4 байти (32 розряди) або зі збільшеною кількістю розрядів мантиси для підвищення точності у форматі слова 8 байтів (64 розряди).

Отже, інформаційну ємність цифрових пристроїв, каналів передачі даних оцінюють логарифмічною мірою кількості інформації в бітах і байтах, а кількість інформації, що зберігається в пам'я'ї ЕОМ, характеризують ємністю адресованої пам'я'ї.

Будь-яка інформація може зображатися в ЕОМ у формі слова (без знака) або цілого числа зі знаком, в останньому випадку 1 біт слова займає знак. Дробові числа зображаються у формі з нерухомою та рухомою комою. Форма з рухомою комою дозволяє зображати числа в дуже широкому діапазоні з приблизно однаковою похибкою, яка визначається довжиною мантиси, але арифметичні операції ускладнюються тим, що їх потрібно виконувати окремо над мантисою і порядком. Тому ця форма використовується частіше в універсальних ЕОМ. Діапазон зображуваних чисел у формі з нерухомою комою значно вужчий і похибка подання малих чисел може неприпустимо збільшуватися. Проте застосуванням відповідних одиниць вимірювання фізичних величин та масштабуванням дані радіотехнічних пристроїв можна вписати до розрядної сітки таким чином, аби уникнути великих похибок зображення. Простота виконання арифметичних дій і через це можливість підвищення швидкодії та ощадливість обладнання зумовили широке застосування форми зображення чисел з нерухомою комою в спеціалізованих, зокрема, радіотехнічних інформаційно-вимірювальних системах.

5.3. Цифрові коди

5.3.1. Загальні відомості

Різноманітну інформацію: цифрові дані, адреси, друкарські символи, команди тощо обчислювальні пристрої сприймають як повідомлення або слова у вигляді двійкових чисел – цифрових кодів. Цифровий код – це сполучення за певними правилами символів 0 та 1 для відображення повідомлення. Сенс кодів тлумачиться в ЕОМ залежно від виконуваної операції, наприклад, як операнди (числа) під час арифметичних дій, як логічні змінні для виконання логічних операцій, як коди літер після команди “Друкувати” тощо.

Мета кодування полягає в тому, аби забезпечити простоту, надійність зберігання та обробки інформації, мінімум обладнання, узгодження із зовнішніми пристроями, високі швидкодії та вірогідність передавання й оброблення повідомлень, полегшити сполучення людини з ЕОМ та ін. За умов протиріччя окремих вимог задовольняють головні з них для конкретної системи, тому й існує багато кодових різновидів.

За формою передавання й приймання по каналах сполучення розрізняють *послідовні* коди (коли розряди надходять послідовно в часі однодротовою лінією) та *паралельні* (коли всі розряди слова передаються одночасно багатодротовою лінією – шиною). Послідовні цифрові коди потребують менше обладнання і більш завадостійкі, тому застосовуються, головним чином, для сполучення з віддаленими об'єктами. У радіотехнічних та обчислювальних пристроях використовуються переважно паралельні коди, бо вони забезпечують значно вищу швидкодію. Природно, існує й комбінована форма кодів, коли довге слово приймається та обробляється за кілька заходів, наприклад, байтами.

З метою уніфікації та спрощення обладнання для виконання арифметичних операцій над числами зі знаком застосовують прямий, обернений або доповняльний *арифметичні* коди. На відміну від арифметичних, *комбінаторні* коди використовую-

ють для позначення літер, знаків та іншої нечислової інформації. Кодуванням цифр однієї СЧ за допомогою цифр іншої утворюються *складені* коди, наприклад, двійково-десяткові. Спеціального застосування в лічильних пристроях, аналого-цифрових перетворювачах набули *відбиті* (рефлексні) коди. Підвищити завадостійкість під час передавання інформації можна за допомогою *завадостійких* кодів, що дозволяють виявляти й виправляти помилки.

5.3.2. Арифметичні коди зображення чисел зі знаком

Прямим кодом є звичайне двійкове число з додаванням знакового біта (рис. 5.6): 0 – для додатних чисел та 1 – для від'є'них:

$$X_{np} = \begin{cases} 0.X, & \text{якщо } X \geq 0; \\ 1.X, & \text{якщо } X < 0. \end{cases}$$

Приклади. 1) $X=+1011_2$; $X_{np}=0.1011$;

2) $X=-1011_2$; $X_{np}=1.1011$.

В ЕОМ подання чисел зі знаковим бітом трактується як мішаний дріб, в якого цей біт є цілою частиною. Це ніяк не впливає на виконання арифметичних дій, тому далі для зручності вважатимемо всі числа цілими.

Обернений або інверсний код додатного числа збігається з його прямим кодом, а від'є'ного (див. рис. 1.6) – утворюється додаванням одиниці в знаковому розряді та оберненням або інвертуванням X (взаємно оберненим заступленням одиниць нулями, а нулів – одиницями) цифрової частини:

$$X_{об} = \begin{cases} 0.X, & \text{якщо } X \geq 0; \\ 1.\overline{X}, & \text{якщо } X < 0. \end{cases}$$

Приклади. 1) $X=+1011_2$; $X_{об}=0.1011$;

2) $X=-1011_2$; $X_{об}=1.0100$.

Природно, для перетворення оберненого коду від'є'ного числа в прямий досить знову проінвертувати цифрову частину,

тобто взаємно замінити в ній нулі та одиниці.

Доповняльний код використовується для виконання арифметичних дій найширше. У десятковій системі числення доповнення n -розрядного числа $X_{10} \in 10^n - |X_{10}|$, наприклад, доповненням -183_{10} буде 817_{10} . Так само й у двійковій системі з огляду на те, що

$$\sum_{i=0}^{n-1} 2^i = 2^n - 1,$$

маємо доповнення n -розрядного числа

$$\begin{aligned} X_{don} &= 2^n - |X_2| = 1 + \sum_{i=0}^{n-1} 2^i - \sum_{i=0}^{n-1} x_i \cdot 2^i = 1 + \sum_{i=0}^{n-1} (1 - x_i) \cdot 2^i = \\ &= 1 + \sum_{i=0}^{n-1} \bar{x}_i \cdot 2^i = 1 + |X_{ob}|. \end{aligned}$$

Таким чином, доповняльний код від'є'ного числа (див. рис. 5.6) утворюється за допомогою одиниці в знаковому розряді, інвертування цифрової частини та додавання одиниці до молодшого розряду, а код додатного числа збігається з прямим кодом:

$$X_{ob} = \begin{cases} 0.X, \text{ якщо } X \geq 0; \\ 1.(\overline{X} + 1) = 1.(X_{ob} + 1), \text{ якщо } X < 0. \end{cases}$$

Не важко переконатись, що після інвертування одиниці з нулями в кінці числа $\dots 1000_2$ маємо $\dots 0111$, а після додавання одиниці до молодшого розряду знов отримуємо в кінці доповняльного коду $\dots 1000$. Тобто практично для утворення доповняльного коду від'є'ного числа ручним способом розглядаємо прямий код від кінця ліворуч (стрілка на рис. 5.6) і лишаємо незмінними всі нулі та першу одиницю, а решту розрядів до розділової крапки інвертуємо. Так само від доповняльного коду переходимо до прямого.

Приклади. 1) $X = +1011_2$; $X_{доп} = 0.1011$; 2) $X = -1011_2$;

$X_{\text{доп}} = 1.0101$; перевірка: $|X_{\text{доп}}| = 2^4 - |X| = 2^4 - 11_{10} = 5_{10}$.

З табл. 5.5, де наведено арифметичні коди деяких чисел, видно, що в прямому та оберненому кодах існує два зображення нуля, які можуть виникати під час виконання арифметичних дій; ці зображення рівноцінні й не спричиняють помилок. У доповняльному коді зображення нуля єдине, внаслідок чого діапазон від'є'них чисел більший на одиницю (доповняльний код найбільшого від'є'ного n -розрядного числа не утворюється від прямого або оберненого кодів, бо їх не існує в розрядній сітці).

Таблиця 5.5

X_{10}	$X_{\text{пр}}$	$X_{\text{об}}$	$X_{\text{доп}}$
+7	0.111	0.111	0.111
...
+1	0.001	0.001	0.001
+0	0.000	0.000	0.000
-0	1.000	1.111	
-1	1.001	1.110	1.111
-2	1.010	1.101	1.110
-3	1.011	1.100	1.101
-4	1.100	1.011	1.100
-5	1.101	1.010	1.011
-6	1.110	1.001	1.010
-7	1.111	1.000	1.001
-8	-	-	1.000

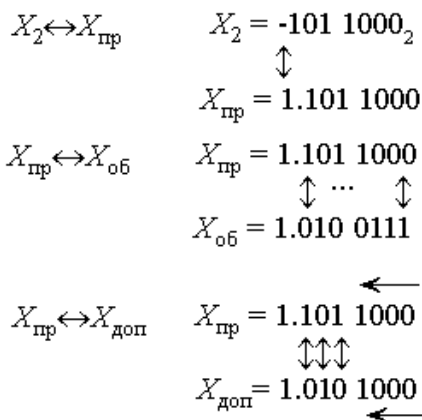


Рис. 5.6. Формування доповняльного коду від'є'ного числа

Отже, числа зі знаком зображаються за допомогою прямого, оберненого та доповняльного арифметичних кодів. Зображення додатного числа у всіх цих кодах однакове: до числа додається нульовий знаковий біт; у зображенні від'є'ного числа знаковий біт дорівнює 1, а в його цифровій частині є модуль числа в прямому коді, інверсія модуля – в оберненому та інверсія плюс одиниця – в доповняльному. Для виконання арифметичних операцій ширше застосовується доповняльний код.

5.3.3. Складені двійково-десяткові коди

Із складених набули поширення лише двійково-десяткові

коди (ДДК), у яких кожна десяткова цифра кодується певним набором двійкових цифр. Це полегшує сполучення з ЕОМ, зокрема, під час введення чисел натисненням на клавішу одразу формується набір нулів та одиниць, які сприймаються ЕОМ, тобто спрощується завантаження даних до ЕОМ у зручній для людини десятковій системі. Якщо завантажені числа використовуються далі для виконання арифметичних операцій, то необхідно перетворити їх у двійкову СЧ або обчислення виконувати безпосередньо в ДДК.

Перший варіант у багатьох випадках є неприйнятний, бо коли виконуються прості арифметичні дії над великими масивами чисел, як, наприклад, в обліковій справі, більша частина машинного часу витрачатиметься на непродуктивні операції перетворення чисел. У другому варіанті використання ДДК з метою спростити виконання арифметичних дій, полегшити декодування, підвищити продуктивність тощо спосіб кодування в ДДК має задовольняти деякі умови. Головними з них є такі: впорядкованість (більшим десятковим цифрам відповідають більші двійкові числа, що утворюють коди, парні цифри – парним двійковим числам), зваженість (кожний розряд коду, як і в двійковій системі, має свою вагу), однозначність (кожній десятковій цифрі відповідає своє двійкове число), самодоповняльність (заступлення в коді кожної цифри нулів одиницями та навпаки утворює нову цифру, що є доповненням вихідної до 9) [12]. Аби задовольнити ці умови чи, принаймні, головні з них, залежно від мети застосовують різні способи кодування.

Найпоширенішим є *ДДК 8421*, частіше його називають просто ДДК (*BCD* – Binary-Coded Decimal – двійково-кодована десяткова система) або код 8421; для стислості позначатимемо його $X_{2/10}$. Кожна десяткова цифра в ДДК кодується двійковою тетрадою з природною вагою розрядів 8, 4, 2, 1 (табл. 5.6), тобто десяткова цифра є просто відповідним їй двійковим чотирирозрядним числом. Тому й перетворення чисел між десятковою системою і ДДК виконуються аналогічно перетворенням між шістнадцятковою та двійковою системами (рис. 5.7), наприклад, $298_{10} = 0010\ 1001\ 1000_{2/10}$. Відміна полягає в тому, що в ДДК набори в тетрадах від 1010 до 1111 є заборонені, бо таких десят-

кових цифр не існує, і всі нулі в тетрадах, включаючи незначущі ліворуч, мають зберігатись, аби уникнути плутанини.

ДДК 2421 або код Айкена (див. табл. 5.6) утворюється аналогічно коду 8421, але відрізняється тим, що вага старшого розряду в ньому дорівнює 2, тому цифри 0...4 збігаються з кодом 8421, а старші утворюються додаванням до десяткової цифри числа 6 та переведенням його в двійковий код:

$$X_{2421} = \begin{cases} X_{2/10}, \text{ якщо } X_{10} < 5; \\ X_{10} + 6_{10} \rightarrow X_2, \text{ якщо } X_{10} \geq 5. \end{cases}$$

ДДК 7421 відрізняється від коду 8421 вагою старшого розряду, що дорівнює 7; цим обумовлено утворення цифр коду (див. табл. 5.6):

$$X_{7421} = \begin{cases} X_{2/10}, \text{ якщо } X_{10} < 7; \\ X_{10} + 1_{10} \rightarrow X_2, \text{ якщо } X_{10} \geq 7. \end{cases}$$

ДДК з надлишком три X' (код 8421+3) утворюється додаванням до десяткової цифри числа 3₁₀ та перетворенням суми у двійкову систему (див. табл. 5.6):

$$X' = X_{8421+3} = X_{10} + 3_{10} \rightarrow X_2.$$

Усі наведені ДДК однозначні, впорядковані за зростанням величини, а перші два з них ще й збігаються за парністю з десятковими цифрами; коди 8421, 2421 та 7421 – зважені, що полегшує виконання арифметичних дій; самодоповняльними є коди 2421 та з надлишком три: $X_0 = \overline{X_9}$, $X_1 = \overline{X_8}$ та ін., що спрощує їх обернення та доповнення в операціях над числами зі знаком.

Код 8421 є поширеніший з огляду на його простоту й відповідність двійковій системі з природною вагою розрядів, але він не є самодоповняльний; коди з надлишком три та 2421 приблизно рівноцінні, недолік першого полягає в тому, що він є незважений, а у другого штучна вага старшого біту тетради. Через зазначені недоліки всі ці коди потребують коригування під час виконання арифметичних дій. Перевага коду 7421 полягає в

зменшенні споживаної потужності від джерела живлення пристроями з динамічними елементами, які не споживають у стані нуля (цей код містить мінімум одиниць); проте його несамоодповняльність та штучна вага старшого розряду призводять до ускладнення арифметичних правил. Існують й інші способи кодування в ДДК, які можуть застосовуватись у цифрових пристроях.

Таблиця 5.6. Основні типи ДДК

X_{10}	8421	2421	7421	8421+3
	8 4 2 1	2 4 2 1	7 4 2 1	
0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 1 1
1	0 0 0 1	0 0 0 1	0 0 0 1	0 1 0 0
2	0 0 1 0	0 0 1 0	0 0 1 0	0 1 0 1
3	0 0 1 1	0 0 1 1	0 0 1 1	0 1 1 0
4	0 1 0 0	0 1 0 0	0 1 0 0	0 1 1 1
5	0 1 0 1	1 0 1 1	0 1 0 1	1 0 0 0
6	0 1 1 0	1 1 0 0	0 1 1 0	1 0 0 1
7	0 1 1 1	1 1 0 1	1 0 0 0	1 0 1 0
8	1 0 0 0	1 1 1 0	1 0 0 1	1 0 1 1
9	1 0 0 1	1 1 1 1	1 0 1 0	1 1 0 0

$X_{10} \leftrightarrow X_{2/10}$

Таблиця 5.7. Формування коду Грея

Код Грея				X_{10}
1	2	3	4	
0	00	000	0000	0
1	01	001	0001	1
	11	011	0011	2
	10	010	0010	3
		110	0110	4
		111	0111	5
		101	0101	6
		100	0100	7
			1100	8
			1101	9
			1111	10
			1110	11
			1010	12
			1011	13
			1001	14
			1000	15

2

9

8_{10}

↕

↕

↕

0010 1001 1000_{2/10}

Рис. 5.7. Формування ДДК числа

5.3.4. Циклічний код Грея

Для того, щоб зменшити вплив збоїв під час переходів в-вхідного коду та уникнути неоднозначності відліку, а також при-

швидшити аналого-цифрове перетворення, застосовують *циклічні* коди, в яких усім сусіднім десятковим числам відповідають кодові набори, відмінні лише одним двійковим розрядом. Звичайному двійковому коду ця властивість не притаманна: під час переходу, наприклад, від 3 до 4 зміна відбувається одразу в трьох двійкових розрядах, а від 7 до 8 – у чотирьох. Проте циклічні коди є непозиційними, тому для виконання арифметичних операцій їх перетворюють у двійковий код.

Завдяки відносній легкості перетворення до двійкового коду і навпаки серед циклічних кодів частіше застосовується код *Грея*, який відноситься до так званих *відбитих* або рефлексних кодів: n -розрядний код дістають відображенням $(n-1)$ -розрядного шляхом відбиття його відносно осі симетрії (у табл. 5.7 показано утворення від одно- до чотирирозрядного коду). Для побудови п'ятирозрядного коду необхідно додати до старшого розряду нуль, а в інші переписати чотирирозрядний код, відтак замінити в старшому розряді нуль на одиницю і відбити в інші розряди знизу дотори чотирирозрядний код.

5.3.5. Завадостійкі коди з виявленням помилок

У радіоелектронних системах, особливо швидкодіючих, внаслідок впливу завад у каналах сполучення та збоїв у роботі пристроїв майже неминуче виникають помилки. Прояв їх полягає у зникненні одиниць або, навпаки, виникненні їх замість нулів в окремих розрядах коду, причому найімовірніше помилка може виникати в одному розряді (одноразова помилка), а одночасне їх виникнення в кількох розрядах має значно нижчу ймовірність. Проте в деяких системах радіозв'язку характерним є виникнення пачок помилок.

Для зменшення чи усунення впливу помилок застосовують *завадостійкі* коди, які дозволяють або тільки виявляти помилки, або виявляти та виправляти їх. Першорядного значення набули коди з виявленням одноразових помилок.

Принципово неможливо виявити, а, отже, і виправити помилки у звичайному двійковому коді, усі комбінації якого є інформаційні. Умовно це показано на рис. 5.8, *а*: деякий кодовий набір, припустимо, тетрада А (на схемі цифрами показано

для прикладу коди тетраді) при зникненні в ній одиниці в будь-якому розряді (показано стрілкою -1) перетворюється в деякий припустимий набір, наприклад *B*, а за виникнення одиниці замість одного з нулів (показано стрілкою +1) – в інший інформаційний код *C*. Так само й інші кодові набори внаслідок одноразової помилки перетворюватимуться один в одного, наприклад, *B* чи *C* в *A*. Це відбувається тому, що мінімальна *кодова відстань* (кількість розрядів, на яку відрізняються два з будь-яких наборів) у двійковому коді дорівнює одиниці: $d=1$. У цьому розумінні коди *A*, *B* та *A*, *C* є *сусідні*.

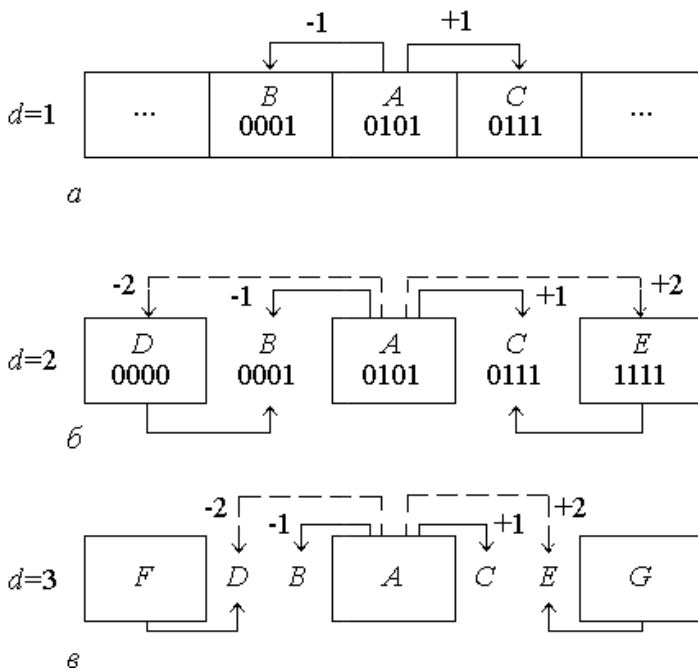


Рис. 5.8. До виявлення помилок у двійковому коді

Для виявлення помилок необхідно забезпечити кодову надлишковість: частина кодових наборів має бути забороненою, а решта – інформаційною, як, наприклад, у ДДК. Проте в ДДК надлишковість недостатня, у ньому можна виявити лише частину помилок. Виявити одноразову помилку можна впевнено в прикладі рис. 5.8, б, при цьому помилка в одному розряді коду *A* типу -1 або

+1 утворює набори В чи С, які в цьому коді є заборонені.

Отже, найближчі інформаційні набори D, E розташовано тепер на кодовій відстані 2, тому, якщо в А є парна кількість одиниць, то в цих наборах вона також буде тільки парною (нуль вважається парним), а в хибних наборах В, С – тільки непарною.

У ДДК можна утворити лише 8 парних комбінацій з 16, тому необхідно збільшити розрядність коду, щоб дістати 10 цифр. Саме таким є самодоповняльний код $3X+2$ (табл. 5.8): дві будь-які цифри X_{10} в ньому відрізняються не менш, ніж двома розрядами.

Таблиця 5.8. Коди із виявленням помилки

X_{10}	8421	8421P	2 з 5	$3X + 2$
	8 4 2 1	8 4 2 1 P	7 4 2 1 0	
0	0 0 0 0	0 0 0 0 0	1 1 0 0 0	0 0 0 1 0
1	0 0 0 1	0 0 0 1 1	0 0 0 1 1	0 0 1 0 1
2	0 0 1 0	0 0 1 0 1	0 0 1 0 1	0 1 0 0 0
3	0 0 1 1	0 0 1 1 0	0 0 1 1 0	0 1 0 1 1
4	0 1 0 0	0 1 0 0 1	0 1 0 0 1	0 1 1 1 0
5	0 1 0 1	0 1 0 1 0	0 1 0 1 0	1 0 0 0 1
6	0 1 1 0	0 1 1 0 0	0 1 1 0 0	1 0 1 0 0
7	0 1 1 1	0 1 1 1 1	1 0 0 0 1	1 0 1 1 1
8	1 0 0 0	1 0 0 0 1	1 0 0 1 0	1 1 0 1 0
9	1 0 0 1	1 0 0 1 0	1 0 1 0 0	1 1 1 0 1

Проте найпростішим кодом із виявленням одноразової помилки є ДДК з перевіркою на парність (код 8421P), який утворюється безпосередньо з ДДК (для порівняння в табл. 5.8 наведено також ДДК) за допомогою додаткового контрольного розряду парності P (Parity – парність).

Залежно від прийнятого правила перевірки до цього розряду додається одиниця в окремих наборах, аби кількість одиниць у будь-якому з них була парною або непарною. Поодинокі помилка (а також непарна їх кількість) викликає зміну парності, що й виявляє помилкову цифру. Слід зазначити, що цей метод виявлення помилок широко використовується не тільки в ДДК, але й в інших кодах. У багатьох мікропроцесорах операція пере-

вірки на парність виконується автоматично й індикуються за допомогою спеціального біта – прапорця (ознаки) парності P .

Так само помилка виявляється у зваженому (крім цифри нуль) ДДК “два з н'я'ьох” (код 74210), в якому кожна десяткова цифра містить дві одиниці; незначущий розряд 0, по суті, виконує функції розряду P (див. табл. 5.8).

Зазначені коди мають мінімальну кодову відстань не менше двох, тому вони виявляють помилку, але для її виправлення потрібно ще визначити, в якому розряді вона виникає. Якщо відстань $d=2$, принципово це зробити неможливо, бо ті ж самі неприпустимі кодові комбінації B , C можуть виникати за одноразової помилки не тільки від коду A , але й від інших інформаційних кодів, зокрема, D , E (суцільні стрілки на рис. 5.8, б). Крім того, при цьому не виявляються й подвійні помилки (пунктирні стрілки), бо вони викликають перетворення інформаційних кодів один в одного.

Розмірковуючи аналогічно, можна дійти висновку, що для виправлення одноразових помилок кодову відстань потрібно збільшити до $d=3$ (рис. 5.8, в). Тепер від інформаційних кодів A , F , G одноразові помилки (суцільні стрілки) утворюють різні неприпустимі набори D , B , C , E , тому за виникнення, наприклад, кодів B чи C ми точно впевнені, що був переданий код A і, отже, можемо відновити його.

Практично це робиться за допомогою *коригувальних* кодів або *кодів Гемінга*. У коді Гемінга, наприклад, для ДДК кожна тетрада кодується сімома бітами, 4 з яких є інформаційні, а 3 – біти контролю парності в певних розрядах коду. Ці розряди дібрано таким чином, що підсумок перевірки на парність певних бітів отриманого повідомлення утворює двійковий код помилкового розряду, інвертуванням якого хибний код виправляється.

Таким чином, шляхом введення надлишковості коду утворюється відстань d не менше трьох, яка дозволяє виправити одноразові помилки. При цьому виявляються також дворазові помилки (пунктир на рис. 5.8, в), проте виправити їх не можна, для цього необхідно ще збільшити відстань d збільшенням надлишковості коду.

5.3.6. Комбінаторні літеро-цифрові коди

Для обслуговування пристроїв введення-виведення інформації, пристроїв її відображення тощо необхідно забезпечити стандартне сполучення ЕОМ із зовнішнім обладнанням, аби не розробляти його для кожної обчислювальної системи. З цією метою впроваджено кілька кодів зображення символічної (літерної) інформації, наприклад, типу *ASCII* (American Standard Code for Information Interchange – Американський стандартний код для обміну інформацією).

У цих стандартах цифри, літери й інші службові символи кодуються, здебільшого, однобайтовим словом, старший розряд якого може приділятися для контрольного біта парності. Наприклад, в *ASCII* літера А має код 0100 0001, знак ? – 0011 1111, символ "П«вернення каретки" » 0000 1101 та ін. Кожна цифра кодується двома тетрадами, причому старша тетрада, яка називається зоною, містить однаковий набір 1111, а молодша відображає власне цифру в ДДК: від 0000 до 1001 (інші шість наборів 1010...1111 використовуються для кодування знаків). Такий формат називається *зоновим* і використовується тільки для введення-виведення інформації.

Для виконання арифметичних дій застосовують *спакований формат*: після введення кожна цифра займає лише одну тетраду, та, крім того, одна тетрада (молодша) приділяється коду знаку числа, а якщо старша лишається вільною, до неї записується 0. Наприклад, у зображенні $-91_{10} = 0000\ 1001\ 0001\ 1101_2$ число займає 2 байти, молодшою тетрадою 1101 кодується знак мінус, а старша тетрада нульова.

5.4. Двійкова арифметика

5.4.1. Загальні відомості

Арифметичні операції над операндами (багатоцифровими числами) виконуються в будь-якій системі числення за правилами дій над однорозрядними операндами a , b . Найпростішою є таблиця дій у двійковій арифметиці (табл. 5.9). У ній зазначено, що під час додавання в деякому розряді чисел двох одиниць утворюється перенесення c (Carry) до старшого розряду, а сума в цьому розряді $s=0$, тобто $1+1=10_2$. Так само й під час відніман-

ня, якщо зменшуване менше за від’ємник, утворюється позика b' ’Воргоу) зі старшого розряду, що дорівнює двом одиницям молодшого, тому різниця в ньому $s' = \text{'}$.

Таблиця 5.9. Дії над двома операндами

$a \ b$	$a+b$ $c \ s$	$a-b$ $b' \ s'$	ab	$a:b$
0 0	0 0	0 0	0	-
0 1	0 1	1 1	0	0
1 0	0 1	0 1	0	-
1 1	1 0	0 0	1	1

011...1 + 1 = 100...0;
 k одиниць k нулів
 100...0 - - = 011...1
 k нулів k одиниць

Таблиця 5.10. Дії над трьома операндами

i	$c_i \ b_i \ a_i$	c_{i+1}	s_i
0	0 0 0	0	0
1	0 0 1	0	1
2	0 1 0	0	1
3	0 1 1	1	0
4	1 0 0	0	1
5	1 0 1	1	0
6	1 1 0	1	0
7	1 1 1	1	1

Слід зауважити, що в багатоцифровому числі з низкою одиниць, розташованих поруч (показано під таблицею), додавання або перенесення до молодшої з них одиниці перетворює всю низку на нулі, а перший нуль ліворуч від неї – на одиницю. І, навпаки, під час віднімання позика одиниці перетворює низку нулів на одиниці, а одиницю ліворуч – на нуль. З урахуванням цього фактично додавання двох чисел зводиться до підсумовування трьох компонентів (табл. 5.10): доданків i -го розряду a_i, b_i та перенесення до цього розряду з попереднього c_i , у підсумку маємо повну суму в цьому розряді s_i і перенесення до старшого розряду c_{i+1} .

5.4.2. Алгебричне додавання в двійковій системі

Як і в десятковій системі X_{10} , у двійковій системі X_2 (рис. 5.9, a) додавання виконується за два заходи: підсумовування розрядів a_i, b_i утворює підрозрядну суму s' а перенесення c_i , додавання яких дає остаточну суму s_i . Апаратно додавання за такою схемою можна реалізувати двома двовходовими півсума-

торами: у першому визначається $a_i + b_i$, у другому $s' + 'i$, або одним тривходовим суматором: $a_i + b_i + c_i$.

Операцію віднімання, аби не застосовувати додаткових пристроїв, виконують шляхом *алгебричного* додавання в *доповняльному* $X_{\text{доп}}$ або *оберненому* $X_{\text{об}}$ кодах, оперуючи зі знаковими бітами як зі звичайними розрядами чисел. Зрозуміло, що модуль суми двох чисел має задовольняти вимогу правильного вибору їх масштабу:

$$|A+B| < 2^n, \quad (5.3)$$

інакше відбудеться переповнення розрядної сітки – сума не вміститься у відведені для неї n розрядів.

Залежно від модулів та знаків доданків можливі чотири випадки обчислення їх суми, які розглянемо на прикладах.

1) що обидва операнди є додатні (рис. 5.9, б), то за умови (5.3) матимемо модуль суми

$$X = A + B = S < 2^n, \quad (5.4, a)$$

а додавання знакових бітів як звичайних розрядів становить знак суми

$$Z = 0 + 0 = 0, \quad (5.5, a)$$

отже, дістанемо в підсумку правильний результат (знаковий біт відокремлено крапкою)

$$Z.X = 0.S. \quad (5.6, a)$$

За невиконання умови (5.3) в (5.4, а) отримаємо перенесення до знакового розряду, тому (5.5, а) дасть $Z = 0 + 0 + 1 = 1$, і в підсумку (5.6, а) матимемо $Z.X = 1.(2^n - S) = 1.S_{\text{доп}}$, тобто дістанемо від'ємну суму за додатних операндів. Це й буде індукувати переповнення розрядної сітки, що автоматично виконується мікропроцесором.

2) Якщо операнди мають різні знаки і додатний більший за модуль від'ємного (рис. 5.9, в), то, враховуючи, що модуль доповняльного коду від'ємного числа X_2 визначається як $2^n - |X_2|$

і що число 2^n дає одиницю перенесення c до знакового розряду, аналогічно (5.4, а...5.6, а) матимемо:

$$X = A + (2^n - |B|) = (A - |B|) + 2^n = S + c > 2^n; \quad (5.4, б)$$

$$\begin{array}{r}
X_{10} \\
+9 \\
+5 \\
+4 \\
+1 \\
\hline
14
\end{array}
\quad
\begin{array}{r}
X_2 \\
+1001 \ a_i \\
+0101 \ b_i \\
+1100 \ S_i' \\
+1 \ c_i \\
\hline
1110 \ S_i
\end{array}
\quad
\begin{array}{r}
X_{10} \\
+9 \\
+5 \\
-4 \\
\hline
\end{array}
\quad
\begin{array}{r}
X_{\text{доп}} \\
+1.0111 \\
+0.0101 \\
\hline
1.1100
\end{array}
\quad
\begin{array}{r}
X_{\text{об}} \\
+1.0110 \\
+0.0101 \\
\hline
1.1011
\end{array}$$

$X_{\text{пр}} = 1.0100; X = -4_{10}$

а

з

$$\begin{array}{r}
X_{10} \\
+9 \\
+5 \\
\hline
14
\end{array}
\quad
\begin{array}{r}
X_{\text{пр}}, X_{\text{доп}}, X_{\text{об}} \\
\quad \quad \quad +0.1001 \\
\quad \quad \quad +0.0101 \\
\hline
\quad \quad \quad 0.1110
\end{array}
\quad
\begin{array}{r}
X_{10} \\
+9 \\
+5 \\
-14 \\
\hline
\end{array}
\quad
\begin{array}{r}
X_{\text{доп}} \\
+1.0111 \\
+1.1011 \\
\hline
1.1.0010
\end{array}
\quad
\begin{array}{r}
X_{\text{об}} \\
+1.0110 \\
+1.1010 \\
\hline
11.0000 \\
\quad \quad \quad \xrightarrow{1} \\
\hline
\quad \quad \quad 1.0001
\end{array}$$

$X_{\text{пр}} = 0.1110; X = 14_{10}$

б

$$X_{\text{пр}} = 1.1110; X = -14_{10}$$

д

$$\begin{array}{r}
X_{10} \\
+9 \\
+5 \\
-4 \\
\hline
\end{array}
\quad
\begin{array}{r}
X_{\text{доп}} \\
+0.1001 \\
+1.1011 \\
\hline
10.0100
\end{array}
\quad
\begin{array}{r}
X_{\text{об}} \\
+0.1001 \\
+1.1010 \\
\hline
10.0011 \\
\quad \quad \quad \xrightarrow{1} \\
\hline
\quad \quad \quad 0.0100
\end{array}$$

$X_{\text{пр}} = 0.0100; X = 4_{10}$

в

$$Z = 0 + 1 + 1 = 10 \rightarrow 0. \quad (5.5)$$

Рис. 5.9. Алгебраїчне додавання у двійковій системі числення

Отже, відкидаючи одиницю перенесення (показано відокремленням) зі знакового розряду, дістанемо правильний підсумок.

$$Z.X=0.S. \quad (5.6, б)$$

Те ж саме маємо під час додавання в оберненому коді, але в підсумку буде на 1 менше, тому згадане перенесення потрібно додати до суми, аби її скоригувати (стрілка на рис. 5.9, в).

3) Якщо операнди різних знаків, але тепер від'ємний за модулем більший, ніж додатний (рис. 5.9, з), то

$$X=(2^n - |A|) + B = 2^n - (|A| - B) = 2^n - |S| = S_{\text{доп}} < 2^n; \quad (5.4, в)$$

$$Z=1+0=1; \quad (5.5, в)$$

$$Z.X=1.S_{\text{доп}}, \quad (5.6, в)$$

тобто дістанемо правильний результат у доповняльному коді. У цьому випадку, як і в попередньому (за різних знаків доданків), переповнення не утворюється, тому й немає потреби його контролювати.

4) Якщо обидва доданки від'ємні (рис. 5.9, д), то за умови (5.3) отримаємо:

$$\begin{aligned} X &= (2^n - |A|) + (2^n - |B|) = 2^n + [2^n - (|A| + |B|)] = \\ &= 2^n + (2^n - |S|) = c + S_{\text{доп}} > 2^n; \end{aligned} \quad (5.4, \text{з})$$

$$Z = 0 + 1 + 1 = \underset{\text{1}}{10} \rightarrow 0. \quad (5.5, \text{з})$$

$$Z.X = 1.S_{\text{доп}} \quad (5.6, \text{з})$$

Як і в другому випадку, відкидаючи перенесення зі знакового розряду під час додавання в доповняльному коді та додаючи його до суми під час додавання в оберненому коді, отримаємо правильний результат відповідно в доповняльному й оберненому кодах. Щодо контролю переповнення, воно виконується аналогічно першому випадку.

Віднімання чисел зі знаком легко замінити їх додаванням, якщо від'ємник знов перетворити в доповняльний або обернений код. Наприклад, операція $(-A) - (-B) = -A + B$ фактично повторює операцію додавання (див. рис. 5.9, з), тому розглянуті правила дійсні й для віднімання.

Отже, щодо додавання чисел з нерухомою комою в двійковій системі слід зазначити таке:

1) додавання чисел зі знаком зводиться до операції додавання в доповняльному або оберненому коді їх розрядів, включаючи знаковий;

2) якщо виникає перенесення в знаковому розряді суми, то під час виконання операції в доповняльному коді його відкидають, а в оберненому коді – додають до суми;

3) результат одержують у тому ж коді, в якому зображено операнди;

4) якщо доданки мають різні знаки, то переповнення не виникає, а якщо однакові, то знак суми збігається з їх знаком, в іншому разі індикується переповнення розрядної сітки;

5) у доповняльному коді не виникає потреби коригувати

суму додаванням до неї одиниці перенесення знакового розряду, тому він використовується частіше;

б) віднімання чисел зі знаком, зображених у доповняльному або оберненому кодах, зводиться до їх додавання, якщо зменшуване залишити незмінним, а над від'ємником виконати операцію відповідно доповнення або обернення.

Для додавання чисел $A = M_1 \cdot 2^{E_1}$ та (де $E_2 < E_1$) у форматі з рухомою комою для одержання суми

$$S = A + B = \left(M_1 + M_2' \right) \cdot 2^{E_1}$$

доданків, збільшуючи менший E_2 до більшого E_1 із коригуванням мантиси M_2 до M_2' , аби число B не змінилося. Відтак алгебрично підсумовують мантиси $M_1 + M_2'$ урахуванням знаків за правилами додавання чисел з нерухомою комою, приписують порядок суми E_1 та, у разі потреби, нормалізують результат.

5.4.3. Алгебричне додавання в системі ДДК 8421

Як і в двійковій системі, алгебричне додавання чисел зі знаком у ДДК виконують у доповняльному або оберненому кодах: операцію доповнення здійснюють над кожною тетрадою до 9 в обох кодах, а над молодшою – до 10 в доповняльному коді. На рис. 5.10 наведено приклади на додавання чисел ДДК 8421 у доповняльному коді: для наочності цифрові розряди десяткового доповняльного коду $X_{10, \text{доп}}$ подано десятковими цифрами, а знакові розряди – двійковими, тобто $+57_{10} = 0.57_{10, \text{доп}}$, $-57_{10} = 1.43_{10, \text{доп}}$ (бо $100 - 57 = 43$). При цьому вважаємо дані коректними: модуль суми операндів не виходить за межі розрядної сітки.

Залежно від величини тетрад доданків можливі три припущення ДДК випадки.

1) Після додавання кожна тетрада суми $S_i < 10$ (рис. 5.10, а), тобто не утворюються неприпустимі комбінації в тетрадах і перенесення між ними: підсумок правильний, коригування його не потрібне.

2) У будь-якій тетраді виникає сума $9 < S_i < 16$ (на рис. 5.10, б – у молодшій тетраді), тобто в ній утворюється заборонена комбінація, але перенесення між тетрадами відсутнє: це при-

зводить до втрати одиниці в наступній, старшій тетраді – потрібне коригування шляхом додавання $6_{10}=0110_2$ до тієї тетради, в якій виникла неприпустима комбінація.

3) Сума двох тетрад данків $S_i > 15$, внаслідок чого утворюється перенесення (на рис. 5.10, в показано стрілкою) з молодшої тетради до старшої (так зване півперенесення *НС* – Naalf Carry, прапорець-ознака якого спеціально для цього вводиться в деяких мікропроцесорах): при цьому молодша тетрада зменшується на 6_{10} – коригується додаванням 6_{10} до тієї тетради, з якої відбулося перенесення. (Перенесення, що виникають під час коригування, повторно не коригуються).

В іншому алгебричне додавання таке саме, як і в двійковій системі: в останній сумі, що утворюється після коригування, перенесення зі знакового розряду відкидається (рис. 5.10, г, д), підсумок одержується в тому ж коді, в якому зображено доданки. Для переведення суми, у разі потреби, з доповняльного в прямий код необхідно знов виконати операцію доповнення: відняти кожну тетраду суми від 9, а молодшу – від 10.

Таким чином, алгебричне додавання в ДДК 8421 вико-

X_{10}	$X_{10, \text{доп}}$	$X_{2/10, \text{доп}}$
$\begin{array}{r} 57 \\ +21 \\ \hline 78 \end{array}$	$\begin{array}{r} 0.57 \\ +0.21 \\ \hline 0.78 \end{array}$	$\begin{array}{r} 0.0101\ 0111 \\ +0.0010\ 0001 \\ \hline 0.0111\ 1000 \\ +\quad 7\quad 8_{10} \end{array}$
<i>a</i>		
X_{10}	$X_{10, \text{доп}}$	$X_{2/10, \text{доп}}$
$\begin{array}{r} 57 \\ +26 \\ \hline 83 \end{array}$	$\begin{array}{r} 0.57 \\ +0.26 \\ \hline 0.83 \end{array}$	$\begin{array}{r} 0.0101\ 0111 \\ +0.0010\ 0110 \\ \hline 0.0111\ 1101 \\ +\quad 0110 \\ \hline 0.1000\ 0011 \\ +\quad 8\quad 3_{10} \end{array}$
<i>б</i>		
X_{10}	$X_{10, \text{доп}}$	$X_{2/10, \text{доп}}$
$\begin{array}{r} 57 \\ +29 \\ \hline 86 \end{array}$	$\begin{array}{r} 0.57 \\ +0.29 \\ \hline 0.86 \end{array}$	$\begin{array}{r} 0.0101\ 0111 \\ +0.0010\ 1001 \\ \hline 0.1000\ 0000 \\ +\quad 0110 \\ \hline 0.1000\ 0110 \\ +\quad 8\quad 6_{10} \end{array}$
<i>в</i>		
X_{10}	$X_{10, \text{доп}}$	$X_{2/10, \text{доп}}$
$\begin{array}{r} 57 \\ +21 \\ \hline 36 \end{array}$	$\begin{array}{r} 0.57 \\ +1.79 \\ \hline 1\ 0.36 \end{array}$	$\begin{array}{r} 0.0101\ 0111 \\ +1.0111\ 1001 \\ \hline 1.1101\ 0000 \\ +\quad 0110\ 0110 \\ \hline 1\ 0\ 0011\ 0110 \\ \leftarrow +\quad 3\quad 6_{10} \end{array}$
<i>г</i>		
X_{10}	$X_{10, \text{доп}}$	$X_{2/10, \text{доп}}$
$\begin{array}{r} 57 \\ +21 \\ \hline 36 \end{array}$	$\begin{array}{r} 0.57 \\ +1.79 \\ \hline 1\ 0.36 \end{array}$	$\begin{array}{r} 0.0101\ 0111 \\ +1.0111\ 1001 \\ \hline 1.1101\ 0000 \\ +\quad 0110\ 0110 \\ \hline 1\ 0\ 0011\ 0110 \\ \leftarrow +\quad 3\quad 6_{10} \end{array}$
<i>д</i>		
X_{10}	$X_{10, \text{доп}}$	$X_{2/10, \text{доп}}$
$\begin{array}{r} 57 \\ +21 \\ \hline 78 \end{array}$	$\begin{array}{r} 0.57 \\ +1.79 \\ \hline 1\ 0.36 \end{array}$	$\begin{array}{r} 0.0101\ 0111 \\ +1.0111\ 1001 \\ \hline 1.1101\ 0000 \\ +\quad 0110\ 0110 \\ \hline 1\ 0\ 0011\ 0110 \\ \leftarrow +\quad 3\quad 6_{10} \end{array}$
$X_{10, \text{пр}} = 1.78$		
<i>е</i>		
X_{10}	$X_{2/10, \text{пр}} =$	
$\begin{array}{r} 57 \\ +29 \\ \hline 86 \end{array}$	$\begin{array}{r} 1.43 \\ +1.79 \\ \hline 1\ 1.22 \end{array}$	$\begin{array}{r} 1.0100\ 0011 \\ +1.0111\ 1001 \\ \hline 10.1011\ 1100 \\ +\quad 0110\ 0110 \\ \hline 1\ 1.0010\ 0010 \\ \leftarrow +\quad 1.0111\ 1000 \\ \hline -\quad 7\quad 8_{10} \end{array}$
<i>д</i>		
X_{10}	X'	
$\begin{array}{r} 57 \\ +29 \\ \hline 86 \end{array}$	$\begin{array}{r} 1000\ 1010 \\ +0101\ 1100 \\ \hline 1101\ 0110 \\ +0\leftarrow 1\leftarrow 0 \\ \hline 1101\ 0011 \\ \leftarrow +\quad 1011\ 1001 \\ \hline 1000\ 0110 \end{array}$	$\begin{array}{r} A' \\ B' \\ Y_i \\ c_{i+1} \\ D_i \\ S'_i \\ S_i \end{array}$
<i>е</i>		

нується тетрадами за правилами двійкової арифметики з наступним коригуванням суми шляхом додавання до тетради числа 6_{10} в двох випадках: 1) коли в ній утворюється неприпустима комбінація; 2) якщо з неї виникло перенесення до старшої тетради. Як і в двійковій арифметиці, перенесення зі знакового розряду в доповняльному коді відкидається, а в оберненому – додається до суми.

5.4.4. Алгебричне додавання в системі ДДК з надлишком 3

В інших ДДК коригування виконується за власними правилами. При цьому в самодоповняльних кодах (з надлишком 3, 2421) легше здійснюються операції доповнення, виявлення перенесень між тетрадами та коригування. Розглянемо особливості додавання в коді X' з надлишком 3 (див. табл. 5.6). Залежно від значень тетрад доданків A, B тут можливі два випадки.

1) Після додавання будь-яка тетрада суми в десятковій системі $A+B<10$ або в коді з надлишком 3: $Y_i=A'+B'=A+3+B+3<16$, тобто не утворюється перенесення до старшої тетради c_{i+1} . У цьому разі матимемо суму $Y_i=(A+B+3)+3=S'+3$, яка перевищує зображення в даному коді S_i на 3, тобто потрібне коригування шляхом віднімання від цієї тетради 3_{10} , що еквівалентно додаванню до тетради доповнення $D_i=2^4-3=13_{10}=1101_2$ з ігноруванням перенесення після коригування, якщо воно виникає.

2) Сума доданків $A+B\geq 10$ або в нашому коді $Y_i=A'+B'=A+3+B+3\geq 16$, внаслідок чого утворюється перенесення до старшої тетради $c_{i+1}=1$, а в підсумку залишається решта суми $Y_i=A'+B'-16=A+3+B+3-16=A+B-10$, що менше потрібної $S_i=(A+B+3)-10$ на 3, тобто необхідно скоригувати тетраду додаванням до неї числа $D_i=3_{10}=0011_2$.

Обидва випадки ілюструються на рис. 5.10, *е*: після підсумовування доданків $A'+B'$ утворюються суми тетрад Y_i та перенесення c_{i+1} (подані стрілками). У прикладі з молодшої тет-

ради перенесення утворюється, а зі старшої – не утворюється, тому ці тетради коригуються на величину D_i відповідно 0011_2 та 1101_2 . Після додавання попередніх сум Y_i , перенесень c_{i+1} та корекцій D_i дістаємо суму S_i' у коді з надлишком 3, відкидаючи перенесення після коригування, якщо вони виникають (на рис. 5.10, *e* відокремлено перенесення зі старшої тетради). Перехід до ДДК 8421 відповідає шуканому підсумку $S_i = 86_{10}$.

Так само виконується й алгебричне додавання в коді з надлишком 3 чисел зі знаком. Проте правила обернення та доповнення через самодоповняльність коду такі ж самі, як і для двійкового коду, що значно спрощує перетворення.

Таким чином, алгебричне додавання в коді з надлишком 3 виконується за такими правилами коригування: якщо з тетради не виникає перенесення до старшої, до неї додається число 13_{10} , а якщо виникає – число 3_{10} , причому при утворенні перенесень під час коригування ними нехтують.

5.4.5. Множення

Під час множення модуль добутку не залежить від знаків співмножників, тому для спрощення алгоритму множення співмножники зображають у прямому коді і окремо визначають знак добутку операцією виключне АБО над знаковими розрядами та його модуль шляхом множення співмножників, вважаючи їх додатними, і, нарешті, до модуля приписують знак.

Обчислення модуля цифровими пристроями виконують способом множення “у стопчик”. У двійковій системі частковий добуток множеного на розряд множника можна дістати зсувом множеного ліворуч (“у«нівський” »посіб) або праворуч. Аби можна було скористатися стандартними суматорами, додавання часткових добутків виконується по черзі.

Але якщо зсувати множене, для його зберігання і зсуву потрібно мати велику кількість розрядів, як і для суми часткових добутків. Для заощадження обладнання множник і множене лишають нерухомими і виконують *зсув попередніх сум* часткових добутків. За такого способу зсувів велика кількість розрядів має виділятися тільки для пристрою, де утворюються суми, отже, і

остаточний модуль добутку. Взагалі, кількість розрядів добутку дорівнює сумі розрядів співмножників або вдвічі довша за їхню розрядність, якщо вони однакової довжини.

Проте під час наближених обчислень, зокрема, дробових чисел, довжину добутку обмежують довжиною співмножників або одного з них, що має більшу похибку. Для цього добуток вкорочують, відкидаючи молодші розряди або округлюють, додаючи до молодшого розряду, що залишається, старший з відкинутих.

Для додаткового скорочення розрядності обладнання зсув *попередніх сум* часткових добутків, починаючи з молодшого розряду, виконують *праворуч* (рис. 5.11, а).

Перший частковий добуток множеного на молодший розряд множника, позначений (1), зсувається праворуч на один крок, якщо наступний розряд множника дорівнює 1, або ще на один крок, якщо дорівнює 0 (у прикладі позначено стрілками). Після операції зсуву і додавання до наступного часткового добутку (2) утворюється попередня сума (під рискою), яка знов зсувається і додається до добутку (3) множеного на старший розряд множника. Остаточна сума є модулем шуканого добутку, до якого долучається ще знаковий біт, знайдений окремо. Якщо в нашому прикладі співмножники є $14_{10}=0.1110_{\text{пр}}$ і $-13_{10}=1.1101_{\text{пр}}$, маємо добуток $-182_{10}=1.10110110_{\text{пр}}$. За необхідністю розрядність сум можна одразу зменшити, нех-

$$\begin{array}{r}
 X_{10} \\
 \times 14 \\
 \hline
 13 \\
 42 \\
 \hline
 + 42 \rightarrow \\
 14 \\
 \hline
 182
 \end{array}
 \quad
 \begin{array}{r}
 X_2 \\
 \times 1110 \\
 \hline
 1101 \\
 1110 \\
 \hline
 + 1110 \\
 \hline
 1110 \\
 1000110 \\
 \hline
 + 1000110 \\
 \hline
 1110 \\
 10110110
 \end{array}
 \quad
 \begin{array}{r}
 (1) \\
 (2) \\
 (3)
 \end{array}$$

а

$$\begin{array}{r}
 + 0.1101110 \mid 0.1010 \\
 + 1.0110 \mid \mid \mid 1011 \\
 (1) \downarrow 0.0011 \mid \mid \mid >0 \uparrow \\
 \leftarrow + 0.0111 \mid \mid \mid \\
 + 1.0110 \\
 (2) \downarrow 1.1101 \mid \mid \mid <0 \uparrow \\
 \leftarrow + 1.1011 \mid \mid \mid \\
 + 0.1010 \\
 (3) \downarrow 0.0101 \mid \mid \mid >0 \uparrow \\
 \leftarrow + 0.1010 \mid \mid \mid \\
 + 1.0110 \\
 (4) \downarrow 0.0000 \mid \mid \mid =0 \uparrow
 \end{array}$$

б

Рис. 5.11. Множення у двійковому коді

туючи молодшими бітами, які під час зсуву виходять за межі розрядної сітки (показана пунктиром).

Існують й інші алгоритми множення, зокрема, у доповняльному або оберненому кодах з одночасним визначенням знаку й модуля добутку. Але через те, що множення довгих чисел потребує багато кроків зсуву і додавання, воно може займати до 80% часу в пристроях обробки інформації і знижує їх швидкість. Якщо це виявляється неприйнятним, особливо в радіотехнічних пристроях, вдаються до використання апаратних та алгоритмічних засобів пришвидшення множення.

Отже, у форматі зображення чисел з нерухомою комою множення зводиться до виконання операцій додавання та зсуву. Найпростішими є алгоритми множення в прямих кодах, коли знак та модуль добутку визначають окремо. Швидше здійснюється множення зсувом множеного, а зменшення вимог до обладнання виникає, якщо зсувати часткові добутки, причому при зсуві їх праворуч виконується одночасно операція округлення.

Множення чисел $A = M_1 \cdot 2^{E_1}$ та $B = M_2 \cdot 2^{E_2}$ у форматі з рухомою комою виконується за правилами множення та додавання чисел з нерухомою комою, при цьому модуль добутку

$$X = M_1 M_2 \cdot 2^{E_1 + E_2}$$

обчислюється множенням модулів мантис $M_1 M_2$, його порядок – алгебричним додаванням порядків співмножників $E_1 + E_2$, а знак – порівнянням знаків чисел A і B . Далі, зазвичай, виконується операція нормалізації мантиси добутку.

Найпростіший спосіб множення в ДДК полягає в додаванні множеного таку кількість разів, яка дорівнює множнику. Для цього множник записується в лічильник, вміст якого декрементується (зменшується на одиницю) під час кожного додавання доти, поки не стане рівним нулю. Отримана сума й буде дорівнювати добутку. З огляду на те, що такий спосіб потребує багато часу, вдаються до різних алгоритмів пришвидшення множення, включаючи дуже складні. Тому майже кожний тип ЕОМ має свій спосіб множення в ДДК.

5.4.6. Ділення

В цифрових пристроях ділення можна виконати аналогічно ручному способу (рис. 5.11, б). У двійковій арифметиці воно полягає у відніманні дільника від діленого або наступних залишків від ділення та зсуву залишків з приписаним черговим розрядом діленого (показано стрілками: зсуваються розряди модуля, а знаковий розряд залишається нерухомим). Якщо різниця після віднімання додатна або дорівнює нулю (показано в дужках: 1, 3, 4), у частку записують одиницю, а якщо різниця від'ємна (2), до частки вносять нуль.

Відмінність машинних методів полягає в тому, що віднімання замінюється алгебричним додаванням, причому після отримання від'ємного залишку (2) в черговому кроці ділення дільник додається не в доповняльному (1.0110), а в прямому коді (0.1010). Крім того, замість зсувати дільник праворуч ("у«нівський" »посіб) з метою заощадження обладнання його залишають нерухомим, а зсувають *залишки ліворуч* (показано стрілками). Процес повторюється до отримання нуля в залишку або здобуття потрібної кількості розрядів.

Таким чином, ділення чисел з нерухомою комою аналогічне множенню: знак визначається так само, а модуль обчислюється виконанням операцій додавання в доповняльному коді та зсуву. Застосовується ділення зі зсувом дільника праворуч (краща швидкодія) або залишків ліворуч (спрощується обладнання). З огляду на те, що операції ділення та множення є сумісні, їх виконують на спільному обладнанні, побудова якого визначається множенням, бо воно зустрічається на порядок частіше ділення.

У формі зображення чисел з рухомою комою ділення також аналогічне множенню з тією різницею, що мантиси ділять як числа з нерухомою комою, а порядки віднімають, тобто додають алгебрично.

Отже, усі арифметичні дії в двійковій системі зводяться до виконання простих операцій додавання двійкових чисел, їх обернення та зсуву за своїми алгоритмами. Це дозволяє уніфікувати й зменшити обладнання арифметичних пристроїв.

Запитання та вправи

1. Як записати число, що дорівнює основі будь-якої позиційної системи числення, цифрами цієї самої системи?

2. Якою кількістю інформації можна охарактеризувати пристрій:

1) здатний відобразити: а) лише одне конкретне число; б) 128; в) 1024; г) 2^{24} різних двійкових чисел?

2) який вміщує: а) 2; б) 12; в) 20; г) 32 двійкові розряди?

3. Скільки різних чисел можна зобразити за допомогою чотирьох: 1) двійкових; 2) вісімкових; 3) десяткових; 4) шістнадцяткових розрядів? Запишіть у десятковій системі максимальне із зображуваних чисел.

4. Що називається машинним словом? В яких одиницях вимірюється його довжина? Що розуміють під терміном “машинний нуль”?

5. Що розуміють під ємністю адресованої пам’яті? Яку ємність адресованої пам’яті спроможна забезпечити система, в якій для адреси виділено: 1) 9; 2) 11; 3) 19; 4) 22 двійкові розряди, якщо в кожній комірці пам’яті зберігається 1 байт інформації?

6. У чому полягають позитивні якості та недоліки двох форм зображення чисел – із нерухомою і рухомою комою?

7. Яка кодова відстань забезпечує: 1) виявлення одноразових помилок; 2) виправлення одноразових помилок; 3) виявлення дворазових помилок; 4) виправлення дворазових помилок?

8. Чим відрізняються зоновий і спакований формати літеро-цифрових кодів? Наведіть приклади.

9. За якими ознаками можна індикувати переповнення розрядної сітки під час алгебричного додавання двійкових чисел? За якої умови переповнення не виникає?

10. Знайдіть X:

1) $3722.76_8 = X_{16}$;

2) $133102.332_4 = X_2$;

3) $7D2.F8_{16} = X_4$;

4) $11111010010.11111_2 = X_8$;

5) $1000110.011_2 = X_{10}$;

6) $46.6_{16} = X_{10}$;

7) $1012.12_4 = X_{10}$;

8) $106.3_8 = X_{10}$;

- | | |
|-----------------------------|--------------------------|
| 9) $250.75_{10}=X_4$; | 10) $500.75_{10}=X_8$; |
| 11) $1000.75_{10}=X_{16}$; | 12) $125.75_{10}=X_2$; |
| 13) $0.F..C_6=X_{10}$; | 14) $1...100_2=X_{10}$; |
| 15) $0.03...3_4=X_{10}$; | 16) $7...70_8=X_{10}$; |
| 17) $1210_4=X_5$; | 18) $244.2_6=X_3$; |
| 19) $400_5=X_4$; | 20) $3201.1_3=X_6$. |

Вказівка: у варіантах 13...16 подати точний розв'язок, користуючись степенями чисел, якщо через три крапки позначено 20 цифр поспіль.

11. Складіть блок-схему алгоритму Горнера перетворення чисел від системи числення з довільною основою до десяткової системи.

12. Розподіліть адреси чотирьох блоків пам'яті ємністю відповідно 32; 8; 2 та 1,5 Кбайт, починаючи з адреси 4000H.

13. Обчисліть діапазони зображуваних чисел (за модулем) у форматі: 1) з нерухомою комою, якщо модуль числа займає 15 біт; 2) з рухомою комою, якщо мантиса та порядок числа займають по 7 біт. Підрахуйте також відносну похибку зображення на кінцях цих діапазонів.

14. Нормалізуйте, де потрібно, пропоновані числа та подайте їх десяткові еквіваленти:

- | | |
|------------------------------------|---------------------------------------|
| 1) $(1,001101 \cdot 2^{-101})_2$; | 2) $(1,001101 \cdot 2^{-110})_2$; |
| 3) $(0,1001101 \cdot 2^{11})_2$; | 4) $(0,001001101 \cdot 2^{-111})_2$. |

15. Зобразіть десяткове число 196 такими двійково-десятковими кодами: 1) 8421; 2) 2421; 3) 7421; 4) 8421 з надлишком 3; 5) 8421 з розрядом парності (8421P); 6) 2 з 5; 7) $3X+2$. У чому полягають особливості цих ДДК, з якою метою вони застосовуються?

16. Виконайте алгебричне додавання за правилами двійкової арифметики 1) у доповняльному та 2) в оберненому кодах таких десяткових чисел: а) $64+63$; б) $79-31$; в) $25-100$; г) $-64-36$, а також у системі ДДК типу: 3) 8421 та 4) з надлишком 3 таких чисел: а) $64+18$; б) $79+18$; в) $64-18$; г) $18-79$; д) $-79-18$; е) $-64-11$; є) $64+11$; ж) $27+59$.

17. Виконайте множення чисел $25 \cdot (-19)$ та ділення $(-182):14$ за правилами двійкової арифметики.

6. ЛОГІЧНІ ОСНОВИ ЦИФРОВОЇ ТЕХНІКИ

6.1. Алгебра логіки

6.1.1. Загальні відомості

Сукупність сигналів на входах цифрового пристрою в кожний момент часу становить певне двійкове число, а сигнал на кожному з його виходів залежить від цієї сукупності за чітко визначеними правилами. У цьому розумінні пристрій є перетворювачем цифрової інформації, його вхідні сигнали – *логічними* (булевими, двійковими) *змінними* (аргументами) x_i , а вихідні – *логічними* (булевими, перемикальними) *функціями*:

$$y_j = f_j(x_1, x_2, \dots, x_m).$$

Природно, логічні змінні та функції як підсумок виконання певної операції над ними можуть набувати в якусь мить часу одного з двох інформаційних значень: логічного 0 (лог. 0) або логічної 1 (лог. 1). З часом значення змінних, а, отже, і функцій змінюються, а якщо якийсь біт інформації, наприклад, на вході x_i залишається весь час сталим, його називають *константою* нуля або одиниці і позначають часто знаком тотожності: $x_i \in 0$ або $x_i \in 1$. Якщо в двох змінних значення змінюються майже одночасно й вони завжди є протилежними, тобто змінні *парафазно* відображають один і той самий сигнал, то одну з них називають *прямою* x_i , а другу – *інверсною* \bar{x}_i .

Абстрагуючись від фізичних процесів у схемі, її елементної бази, зручно функціонування цифрових пристроїв та моделювання інформаційних перетворень у них розглядати за допомогою двоцифрової *алгебри логіки* (булевої алгебри). Для цього, передусім, потрібно якимось чином визначити залежність кожного вихідного сигналу від вхідних, тобто задати його логічну функцію. Первісними є словесний і табличний *способи задання функцій*, з яких вже можна дістати її зображення у вигляді логічного (булевого) виразу. Практично найчастіше

використовується задання за допомогою *таблиці відповідності* (істинності, перемикальної). Зручною є така її побудова (див., наприклад, табл. 5.10), коли кожна змінна і функція розташовані в колонку; одночасно змінні утворюють рядки – *вхідні кортежі* (набори вхідних змінних, слова вхідної абетки), які є їх двійковими відповідниками. Якщо змінні розташувати в природному порядку, як і двійкові числа – від старшого розряду числа до молодшого, вони займатимуть stále положення незалежно від їх кількості на графічній моделі – діаграмі, що зручно під час синтезу функцій. Крім того, літерою *i* часто позначатимемо десятковий код вхідного кортежу для зручності посилань.

З усіх $k=2^m$ кортежів – можливих наборів змінних від 0 до 2^m-1 (де m – кількість змінних) – можна утворити 2^k різних функцій. Деякі з них є *виродженими* – це константи нуля чи одиниці і такі, що повторюють одну із змінних або функцію від меншої кількості аргументів, а деякі не є самостійними, якщо відрізняються лише порядком розташування змінних. Тому далі розглядаються лише дві функції однієї змінної (одномісні операції) та 4 пари оригінальних функцій двох змінних (двомісні або бінарні операції); до трьох з цих пар зводяться функції і більшої кількості аргументів. Тут і далі користуватимемося, здебільшого, лаконічними термінами алгебри логіки, а паралельні назви – літературні варіанти – подаватимемо в дужках.

6.1.2. Основні логічні функції

1. Означення. У табл. 6.1 усі функції (в дужках англійською мовою зазначено їх скорочені назви, що використовуються під час програмування) подано парами: у лівих вихідних колонках таблиці відповідності та умовних графічних позначень наведено пряму функцію y_0 , а в правих – інверсну y_1 .

Лише одна з наведених операцій – *функція повторення* (теж, твердження) $y_0=x_1$ (читається: як x_1) є тривіальна, проте повторювач або буферний підсилювач, що її виконує, часто використовується для підвищення навантажувальної здатності цифрових пристроїв та сполучення їх із шинами. Друга одномісна операція – *функція НІ* (логічне заперечення, інверсія,

доповнення) означає обернення аргументу x_1 , тобто зміну його логічного значення на протилежне. Інверсія позначається у формулах рискою над змінними: $y_1 = \bar{x}_i$ (читається: не x_1), а на умовних графічних позначеннях інвертора (елемента НІ) та інших елементів – маленьким колом.

Функція АБО (логічне додавання, логічна сума, диз'юнкція, об'єднання) – це операція збирання інформації від кількох джерел до одного каналу: на виході y_0 з'являється рівень лог. 1, якщо він надходить хоча б з одного з входів. Логічне додавання відрізняється від арифметичного останнім рядком таблиці: $1+1=1$, позначається $y_0 = x_1 + x_2 = x_1 \vee x_2$ (читається: x_1 або x_2), реалізується елементом АБО (елементом збирання, диз'юнктором). З інверсією на виході утворюється *функція АБО-НІ* (логічне додавання із запереченням, стрілка Пірса, заперечення диз'юнкції, доповнення об'єднання): $y_1 = \overline{x_1 \vee x_2} = \overline{x_1} + \overline{x_2}$ (читається: ні x_1 , ані x_2), виконується елементом АБО-НІ (елементом Пірса).

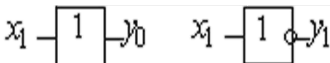
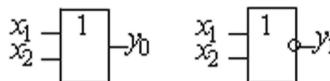
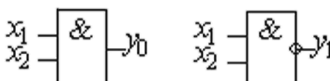
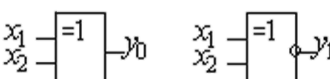
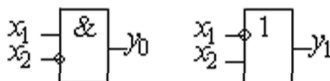
Функція І (логічне множення, добуток, кон'юнкція, перетин) – операція визначення збігу інформації на всіх входах; вона не відрізняється від двійкового множення одноцифрових чисел: набуває значення лог. 1, якщо високий рівень збігається на всіх входах. У виразах позначається $y_0 = x_1 x_2 = x_1 \cap x_2$ (читається: x_1 і x_2), а на умовних позначеннях – знаком & (амперсанд, and). Виконується операція елементом І (елементом збігу, кон'юнктором). Симетрична їй функція *І-НІ* (логічне множення із запереченням, риска Шеффера, заперечення кон'юнкції, несумісність): $y_1 = \overline{x_1 x_2} = \overline{x_1} \cap \overline{x_2}$ (читається: не x_1 або не x_2), реалізується елементом І-НІ (елементом Шеффера). Логічні функції АБО, І та їх заперечення можуть виконуватися над довільною кількістю змінних.

Функція виключне АБО (додавання або сума за модулем 2, нерівнозначність, антиеквівалентність) відрізняється від функції АБО тим, що під час додавання кожної пари одиниць утворюється нуль; найчастіше операцію над багатьма змінними називають додаванням за модулем 2, а над двома аргументами – виключним АБО. Позначається функція за допомогою символа

псевдо-плюс: $y_0 = x_1 \oplus x_2 = \overline{x_1 x_2} + \overline{x_1} x_2$ (читається: або x_1 , або x_2 ; x_1 або x_2 виключно), а на графічних зображеннях – символом *M2* та (для двовходових елементів) позначкою =1. Елемент, що її реалізує, з довільною кількістю входів звичайно називають суматором за модулем 2, а двовходовий, найчастіше, - елементом виключне АБО (елементом нерівнозначності). Суміжна функція *виключне АБО-НІ* (рівнозначність, еквівалентність):

$y_1 = x_1 \oplus x_2 = x_1 \sim x_2$ (читається: x_1 як x_2) виконується цифровим пристроєм, який теж називається елементом виключне АБО-НІ (елементом рівнозначності, еквівалентором).

Таблиця 6.1. Основні типи логічних функцій та логічних елементів

Логічна функція		Таблиця відповідності		Умовне графічне позначення	
Назва	Позначення	$x_2 x_1$	$y_2 y_1$		
Повторення	$y_0 = x_1$	0	0 1		
НІ (NOT)	$y_1 = \overline{x_1}$	1	1 0		
АБО (OR)	$y_0 = x_1 + x_2$	0 0	0 1		
АБО-НІ (NOR)	$y_1 = \overline{x_1 + x_2}$	0 1	1 0		
		1 0	1 0		
		1 1	1 0		
І (AND)	$y_0 = x_1 x_2$	0 0	0 1		
І-НІ (NAND)	$y_1 = \overline{x_1 x_2}$	0 1	0 1		
		1 0	0 1		
		1 1	1 0		
Виключне АБО (XOR)	$y_0 = x_1 \oplus x_2$	0 0	0 1		
Виключне АБО-НІ (XNOR)	$y_1 = \overline{x_1 \oplus x_2}$	0 1	1 0		
		1 0	1 0		
		1 1	0 1		
Забора	$y_0 = x_1 \setminus x_2$	0 0	0 1		
Імплікація	$y_1 = x_1 \rightarrow x_2$	0 1	1 0		
		1 0	0 1		
		1 1	0 1		

Функція заборона (збіг із заборonoю, заперечення імплікації, різниця) означає, що вхідна змінна x_1 надходить до виходу лише за відсутності заборони x_2 (якщо $x_2=0$), інакше на виході є рівень лог. 0. Найчастіше під цією функцією розуміють логічне множення з інверсією по одному з двох входів: $y_0 = x_1 \setminus x_2 = \overline{x_1} x_2$ (читається: x_1 , але не x_2 ; x_1 і не x_2). Проте інколи її зручно тлумачити як своєрідне логічне віднімання $x_1 \setminus x_2 = x_1 - x_2$: з колонки x_1 вилучаються одиниці (якщо вони є) стовпця x_2 . Реалізується вона елементом заборони (елементом І з інверсним входом).

Функція імплікація (розділення із заборonoю, селекція) отримала назву з логіки висловлень через зв'язку “якщо $x_1=1$, то $y_1=x_2$ ” (інакше $y_1=1$), проте часто має тлумачення логічного додавання з інверсією одного з двох входів (або функції заборони із запереченням по виходу): $y_1 = x_1 \rightarrow x_2 = \overline{x_1} + x_2$ (читається: якщо x_1 , то x_2 ; не x_1 або x_2). Елемент, що реалізує таку функцію, називається імплікатор (селектор).

2. Функціонально повна система. Не всі перелічені логічні функції є незалежні. Такий їх набір, за допомогою якого можна утворити логічну функцію будь-якої складності, становить *функціонально повну систему* або *базис*. Система функцій НІ, АБО, І утворює *булів базис*.

Проте й ця система є надлишкова: вона лишається повною й у тому разі, якщо вилучити з неї один з елементів, крім НІ. *Мінімально повною* є система, з якої не можна вилучити жодної функції без утрати її повноти. Можна утворити функціонально повні системи, у тому числі й мінімальні, з різних наборів функцій, але обов'язково в них має бути інверсія. Практичного значення крім булевого набули базиси з двох елементів: 1) І, НІ та 2) АБО, НІ, а частіше навіть з одного елемента: 1) І-НІ та 2) АБО-НІ; через це елементи І-НІ та АБО-НІ називають *універсальними*. Проте для зменшення кількості елементів, необхідних для побудови пристрою, забезпечення гнучкості проектування використовують і складніші функції.

Отже, алгебра логіки оперує з логічними або двійковими змінними й функціями, які можуть набувати двох значень: 0 або 1. Перетворення в алгебрі логіки з метою аналізу, синтезу, дове-

дення положень тощо здійснюють, користуючись, головним чином, основними логічними функціями НІ, АБО, І, але під час практичної реалізації цифрових пристроїв широко застосовують елементи, що виконують комбіновані функції І-НІ, АБО-НІ, І-АБО-НІ, виключне АБО тощо.

6.1.3. Співвідношення алгебри логіки

Основні співвідношення (у літературі існують також їх назви: аксіоми або тотожності, закони або теореми, наслідки законів або правила) розглядаються в булевій алгебрі, здебільшого, відносно функцій АБО та І, а справедливості тих чи інших формул відносно інших функцій з'ясовується окремо. У табл. 6.2 формули пронумеровано цифрами, що відповідають назвам співвідношень, і літерами a , b для функцій АБО та І, а з літерою c для прикладу наведено аналогічні співвідношення для функції виключне АБО.

1. Аксіоми 1...5 безпосередньо впливають з таблиць відповідності основних функцій і часто застосовуються в практиці (на рис. 6.1 нумерація формул зазначена над рисунками в дужках).

Подвійне заперечення 1 використовується для відновлення змінної x , наприклад, у схемах збільшення коефіцієнта розгалуження (рис. 6.1, a) за недостатньої навантажувальної здатності елемента. На рис. 6.1, b , c , d наведено принципи побудови інверторів за допомогою аксіом; ті ж самі принципи застосовуються й для підімкнення зайвих входів мікросхем (підімкнення за аксіомами 2 a , b менше навантажує джерело сигналу).

Крім того, у мікропроцесорах аксіоми застосовуються для програмованої зміни слова або окремих його бітів (на рис. 6.1, d у дужках зазначено, яка операція має міститись для цього в коді відповідної команди). У наведеному прикладі у вихідному слові $96_{2/10}$, розряди якого підлягають зміні, за допомогою дібраного з цією метою другого операнду $0F_{16}$ (його вводять до складу команди) можна встановити одиниці в молодшій тетраді без зміни старшої виконанням операції АБО, обнулити старшу тетраду без зміни молодшої виконанням операції І та

проінвертувати розряди молодшої тетради без зміни старшої операцією виключне АБО.

Дозвіл або блокування проходження змінної на підставі аксіом 2 а, б і 3 а, б широко використовується для синхронізації пристроїв (рис. 6.1, е). Для блокування змінних x_1, x_2 на спільних входах елементів АБО-НІ встановлюють рівень лог. 1, а елементів І-НІ – лог. 0. Тоді, незалежно від сигналів на інформаційних входах x_1, x_2 на виходах елементів буде постійний рівень відповідно лог. 0 та лог. 1. З надходженням активного рівня синхросигналу відповідно $\bar{C}=0$ або $C=1$ змінні x_1, x_2 з'являються на виходах (з інверсією).

Таблиця 6.2. Аксіоми та закони алгебри логіки

Співвідношення	АБО (а)	І (б)	Виключне АБО (в)
Аксіоми:			
1. Подвійне заперечення	$\bar{\bar{x}}=x$		
2. Дозвіл	$x + 0 = x$	$x \cdot 1 = x$	$x \oplus 0 = x$
3. Блокування (інвертування)	$x + 1 = 1$	$x \cdot 0 = 0$	$(x \oplus 1 = \bar{x})$
4. Повторення	$x + x = x$	$x \cdot x = x$	$x \oplus x = 0$
5. Доповнення	$x + \bar{x} = 1$	$x \cdot \bar{x} = 0$	$x \oplus \bar{x} = 1$
Закони:			
6. Переставний	$x_1 + x_2 = x_2 + x_1$	$x_1 x_2 = x_2 x_1$	$x_1 \oplus x_2 = x_2 \oplus x_1$
7. Сполучний	$x_1 + x_2 + x_3 = x_1 + (x_2 + x_3)$	$x_1 x_2 x_3 = x_1 (x_2 x_3)$	$x_1 \oplus x_2 \oplus x_3 = x_1 \oplus (x_2 \oplus x_3)$
8. Розподільчий	$x_1 (x_2 + x_3) = x_1 x_2 + x_1 x_3$	$x_1 + x_2 x_3 = (x_1 + x_2)(x_1 + x_3)$	$x_1 (x_2 \oplus x_3) = x_1 x_2 \oplus x_1 x_3$
9. Двоїстості (де Моргана)	$x_1 + x_2 = \overline{\bar{x}_1 \bar{x}_2}$ $\bar{x}_1 + \bar{x}_2 = \overline{x_1 x_2}$	$x_1 x_2 = \overline{\bar{x}_1 + \bar{x}_2}$ $\bar{x}_1 \bar{x}_2 = \overline{x_1 + x_2}$	
Наслідки:			
10. Склеювання	$x_1 x_2 + x_1 \bar{x}_2 = x_1$	$(x_1 + x_2)(x_1 + \bar{x}_2) = x_1$	$x_1 x_2 \oplus x_1 \bar{x}_2 = x_1$
11. Поглинання	$x_1 + x_1 x_2 = x_1$	$x_1 (x_1 + x_2) = x_1$	$x_1 \oplus x_1 x_2 = x_1 \bar{x}_2$
12. Заступлення	$x_1 + \bar{x}_1 x_2 = x_1 + x_2$	$x_1 (\bar{x}_1 + x_2) = x_1 x_2$	$x_1 \oplus \bar{x}_1 x_2 = x_1 + x_2$

2. Закони. В алгебрі логіки широко використовуються чотири закони 6...9 щодо виконання логічних операцій. Пере-

ставний закон 6 свідчить про рівнозначність інформаційних входів логічних елементів АБО, І, виключне АБО. Проте, природно, закон не виконується для елементів заборони та імплікатора, бо один з їх входів інверсний і його позначення має чітко відрізнятися від прямого. Сполучний закон 7 використовується з метою розширення кількості змінних в операціях АБО, І, виключне АБО шляхом драбинчастого каскадування елементів на кшталт рис. 6.1, *є*, якщо в елементі бракує входів, тобто для збільшення коефіцієнта об'єднання входів.

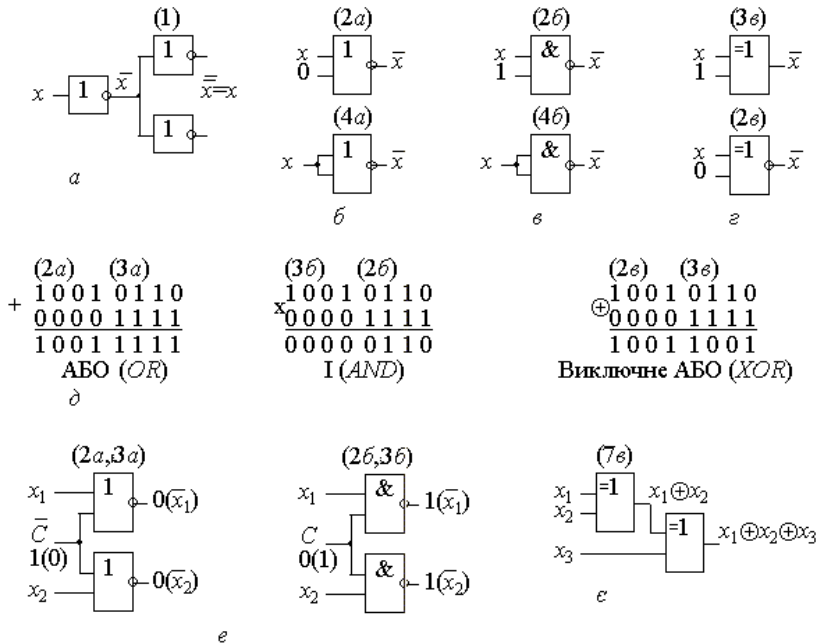


Рис. 6.1. Використання аксіом та законів алгебри логіки

Розподільчий закон 8 дозволяє скоротити кількість елементів, потрібних для виконання логічних функцій (рис. 6.2, *а*, *б*). Якщо для функції АБО він є звичний, то для функції І має форму, відсутню в звичайній алгебрі. Перевірити формулу 8б, як й інші співвідношення, можна методом прямої індукції: підставляючи у формулу всі можливі комбінації змінних, наприклад, за таблицею відповідності, переконуємося в рівності лівої

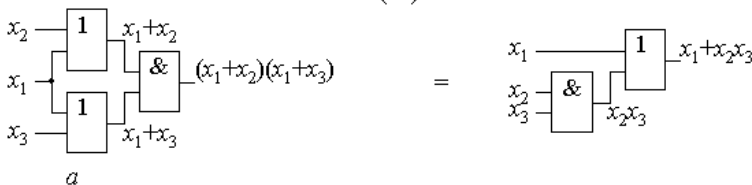
та правої частин виразу для кожного набору змінних.

Інший шлях доведення (який застосовується також для спрощення функцій) – методом тотожних перетворень зводимо одну частину рівності до іншої. Для стислості будемо під елементами виразу зазначати нумерацію співвідношень, на підставі яких зроблені перетворення. Для розподільного закону 8б маємо:

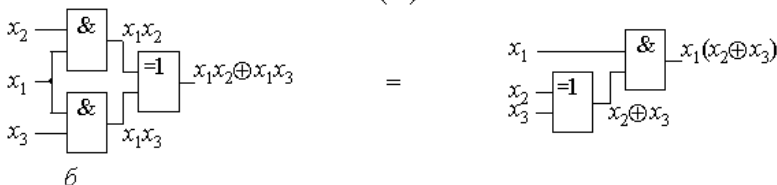
$$\begin{aligned}
 x_1 + x_2 x_3 &= x_1 \cdot 1 + x_2 x_3 = x_1 (1 + x_2 + x_3) + x_2 x_3 = (x_1 + x_1 x_2 + x_1 x_3) + x_2 x_3 = \\
 &\quad \begin{matrix} 2б \\ x_1 x_1 + x_1 x_2 + x_1 x_3 + x_2 x_3 = (x_1 + x_2) \cdot (x_1 + x_3), \end{matrix} \\
 &\quad \begin{matrix} 3а \\ 4б \end{matrix} \qquad \begin{matrix} 8а \end{matrix}
 \end{aligned}$$

що й треба було довести.

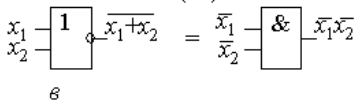
(8б)



(8а)



(9а)



(9б)

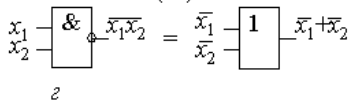


Рис. 6.2. Використання аксіом та законів алгебри логіки

Закони двоїстості 9 (закони інверсії, правила або закони де Моргана) для зручності записано в двох формах на підставі аксіоми 1. Із цих законів випливає двоїстість елементів АБО та І (рис. 6.2, в, г). У *позитивній логіці*, якою ми користуємось за заумовчанням (вважаємо, що завжди лог. 1 кодується високим рівнем напруги, а лог. 0 – низьким), елементи АБО та І викону-

ють однойменну логічну функцію, проте в *негативній логіці* (лог. 1 – відповідає низькому рівню напруги, а лог. 0 – високому), навпаки, елемент АБО можна вважати елементом збігу нулів на його входах (при цьому на виході буде теж нуль), а елемент І – елементом збирання нульових сигналів (на його виході є нуль, якщо хоча б один з аргументів нульовий). Із законів де Моргана можна зробити корисне узагальнення:

$$y = x_1 \bar{x}_2 + x_3 \bar{x}_4; \quad \bar{y} = (\bar{x}_1 + x_2)(\bar{x}_3 + x_4), \quad (6.1)$$

тобто *взаємна заміна* в логічному виразі будь-якої функції у прямих і інверсних змінних та знаків логічного додавання і множення спричиняє її *заперечення*. Наприклад,

$$y = \overline{(x_1 + x_2)(\bar{x}_1 + \bar{x}_2)} = \bar{x}_1 x_2 + x_1 \bar{x}_2.$$

3. Наслідки. Слід відзначити, що закони 6...9 дійсні для довільної кількості аргументів. Це стосується й *наслідків* 10...12, які часто використовуються для спрощення функцій, їх синтезу, інших тотожних перетворень. Проте процес перетворень залежить від індивідуального підходу, досвіду тощо. Про це свідчить, наприклад, доведення наслідку 12а двома шляхами:

$$\begin{aligned} 1) \quad y &= x_1 + \bar{x}_1 x_2; \\ \bar{y} &= \bar{x}_1 (\bar{x}_1 + x_2) = \bar{x}_1 \bar{x}_1 + \bar{x}_1 x_2 = \bar{x}_1 \bar{x}_2 = \bar{x}_1 + x_2; \quad y = x_1 + x_2; \\ &\quad (2.1) \qquad (8a) \qquad (5b) \qquad (9a) \qquad 1 \\ 2) \quad y &= x_1 + \bar{x}_1 x_2 = (\bar{x}_1 + x_1)(x_1 + x_2) = 1 \cdot (x_1 + x_2) = x_1 + x_2. \\ &\quad (8b) \qquad (5a) \qquad (2b) \end{aligned}$$

Таким чином, перетворення логічних функцій та схем виконуються за розглянутими співвідношеннями алгебри логіки. На відміну від звичайної, у булевій алгебрі всі співвідношення симетричні відносно функцій логічного додавання та множення. Аби запобігти помилок щодо послідовності виконання операцій, слід діяти за погодженням: найстаршою є операція логічного заперечення (бо заперечення суми аргументів не дорівнює сумі їх заперечень), за нею йде логічне множення і найнижчою є операція логічного додавання. Послідовність дій можна змінити, як звичайно, за допомогою дужок. З огляду на аксіоми 4 а, б в алгебрі логіки множення на коефіцієнти, відмінні від 0 та 1, а

також піднесення до степеня не мають сенсу.

Тотожні перетворення, особливо складних логічних виразів, тобто їх спрощення, доцільно виконувати формалізованими методами й лише після цього, у разі потреби, завершувати ручним способом.

6.1.4. Стандартні форми логічних функцій

1. Означення. Шляхом тотожних перетворень за законами алгебри логіки можна утворити безліч еквівалентних булевих виразів будь-якої функції. Аби вибрати з-поміж них оптимальну та алгоритмізувати процес перетворень, вдаються до надання логічним виразам стандартної форми.

Для прикладу розглянемо логічну функцію, зображену в таких еквівалентних формах:

$$y_0 = x_1 \oplus x_2 \rightarrow (x_2 \setminus x_3);$$

$$y_1 = x_1 \overline{x_2} + x_2 (\overline{x_1} + \overline{x_3});$$

$$y_2 = x_1 \overline{x_2} + x_2 \overline{x_1 x_3};$$

$$y_3 = x_1 \overline{x_2} + \overline{x_1 x_2} + x_2 \overline{x_3};$$

$$y_4 = x_1 \overline{x_2} \overline{x_3} + x_1 \overline{x_2} x_3 + x_1 \overline{x_2} \overline{x_3} + \overline{x_1} \overline{x_2} x_3 + \overline{x_1} \overline{x_2} \overline{x_3}$$

$$y_5 = (x_1 + x_2 + x_3) (\overline{x_1} + \overline{x_2} + \overline{x_3})$$

Вирази y_0, y_1 подано в нестандартній, мішаній формі, бо в стандартній формі вираз має містити лише елементи булевого базису у вигляді або суми добутків (диз'юнкції кон'юнкцій), або добутку сум (кон'юнкції диз'юнкцій). Саме такою є функція y_2 , зображена в *диз'юнктивній* формі (ДФ), тому що вона є диз'юнкцією над елементами виразу (*термами*), кожний з яких містить тільки операції кон'юнкції та інверсії. На відміну від неї, функція y_3 подана в *диз'юнктивній нормальній* формі (ДНФ), бо її терми містять операції інверсії над окремими змінними, але не над виразами. І, нарешті, зображена функція y_4 є досконалою диз'юнктивною нормальною формою (ДДНФ), яка являє собою диз'юнкцію термів, кожний з яких є кон'юнкція над усіма

літералами.

Згідно з дуальністю алгебри логіки аналогічно можна зобразити функцію в кон'юнктивній (КФ), кон'юнктивній нормальній (КНФ) та досконалій кон'юнктивній нормальній формі (ДКНФ). ДКНФ функції являє собою кон'юнкцію термів, кожний з яких є диз'юнкція над усіма літералами, наприклад, як у виразі y_5 .

З огляду на те, що одна й та сама змінна не може бути поданою в жодному термі двічі (згідно з аксіомами це не має сенсу), логічна функція в ДДНФ або ДКНФ має тільки *єдине зображення*, тому вона є зручною стандартною моделлю репрезентування функції у формалізованих методах перетворень.

Таблиця 6.3. Форми логічних функцій

x_1	x_2	y	\bar{y}	Мінтерми M_i		Макстерми M_i'	
				$y=1$ за умови	$\bar{y}=1$ за умови	$y=0$ за умови	$\bar{y}=0$ за умови
0	0	0	1	$M_1=\bar{x}_1x_2$ $M_2=x_1\bar{x}_2$	$M_0=\bar{x}_1\bar{x}_2$ $M_3=x_1x_2$	$M_0'=x_1+x_2$ $M_3'=\bar{x}_1+\bar{x}_2$	$M_1'=x_1+\bar{x}_2$ $M_2'=\bar{x}_1+x_2$
0	1	1	0				
1	0	1	0				
1	1	0	1				
Стандартні форми				ДДНФ		ДКНФ	
				$y=x_1\bar{x}_2+\bar{x}_1x_2$	$\bar{y}=x_1x_2+\bar{x}_1\bar{x}_2$	$y=(x_1+x_2)*(\bar{x}_1+\bar{x}_2)$	$\bar{y}=(x_1+\bar{x}_2)*(\bar{x}_1+x_2)$

2. Мінтерми й макстерми. Розглянемо для прикладу функцію, задану таблицею відповідності (табл. 6.3). Зобразимо її спочатку за допомогою добутків усіх літералів. Виходимо з того, що на будь-якому входному кортежі функція як добуток літералів набуває значення $y=1$ лише одним способом: до добутку входять змінні без інверсії, якщо в цьому кортежі вони мають значення 1, і з інверсією, якщо їх значення дорівнює 0. Такі добутки називають *мінтермами* (конституентами одиниці); у прикладі їх два: $M_1=\bar{x}_1x_2$, $M_2=x_1\bar{x}_2$. Отже, мінтерми утворюються за одиничними значеннями функції як добутки літералів

$$M_j = \bigwedge_{i=1}^m \tilde{x}_i,$$

де \tilde{x}_i – літерал, m – кількість змінних.

Через те, що кожний мінтерм перетворює функцію на одиницю тільки на своєму кортежі, а на всіх інших кортежах він дорівнює нулю, усю функцію можна зобразити як логічне додавання мінтермів:

$$y = \bigvee_{j=1}^k M_j = \bigvee_{j=1}^k \bigwedge_{i=1}^m \tilde{x}_i,$$

де k – кількість мінтермів, тобто одиниць у колонці функції її таблиці відповідності. У прикладі маємо відомий вираз для функції виключне АБО

$$y = x_1 \oplus x_2 = \overline{x_1 x_2} + \overline{x_1 x_2}, \quad (6.2)$$

зображеної в ДДНФ.

Розмірковуючи аналогічно, з таблиці легко знайти також вираз для інверсної функції в ДДНФ за нулевими її значеннями (тобто одиницями в колонці y):

$$\overline{y} = \overline{x_1 \oplus x_2} = x_1 x_2 + \overline{x_1 x_2}. \quad (6.3)$$

Так само можна зобразити функцію в ДКНФ, якщо виходити з добутку сум літералів. Диз'юнкція літералів, де функція перетворюється на нуль, називається *макстермом* (конституентною нуля), тому до макстерма змінна входить без інверсії, якщо її значення в кортежі дорівнює нулю, інакше – з інверсією. У прикладі є два макстерми: $M_0' = x_1 + x_2$, $M_3' = \overline{x_1} + \overline{x_2}$. Або взагалі

$$M_j' = \bigvee_{i=1}^m \tilde{x}_i,$$

а вся функція зображається в ДКНФ через кон'юнкцію макстермів

$$y = \bigwedge_{j=1}^k M_j' = \bigwedge_{j=1}^k \bigvee_{i=1}^m \tilde{x}_i,$$

де k – кількість макстермів, тобто нулів у колонці функції її таблиці відповідності. Отже, у прикладі маємо ДКНФ

$$y = x_1 \oplus x_2 = (x_1 + x_2)(\overline{x_1 + x_2}) \quad (6.4)$$

Аналогічно можна знайти вираз для інверсної функції в ДКНФ за одиничними значеннями (тобто нулями в її колонці):

$$\overline{y} = \overline{x_1 \oplus x_2} = (x_1 + x_2)(\overline{x_1 + x_2}). \quad (6.5)$$

Таким чином, будь-яка логічна функція зображається в ДДНФ або ДКНФ єдиним способом, тому ці форми є вихідними для подальшого аналізу й синтезу. Інші стандартні форми ДНФ і КНФ можуть бути простішими, ніж відповідні досконалі форми, проте в них функції зображаються багатьма способами, що ускладнює алгоритмізацію їх перетворень. Принципово обидві досконалі форми рівноцінні, але ДДНФ є зручніша для побудови пристроїв у базисі І-НІ, а ДКНФ – у базисі АБО-НІ. Через більшу поширеність елементів І-НІ у складі сучасних мікросхем, а також більш звичні співвідношення алгебри логіки (формули з літерою *a* в табл. 6.2), частіше для зображення функцій використовується ДДНФ.

3. Перехід до досконалих форм від таблиці відповідності. Під час логічного проектування доводиться користуватися різними формами зображення функціонування ЦП: таблиці відповідності, вирази логічної функції того чи іншого типу, діаграми й ін. Ці форми щільно пов'язані між собою і можуть розглядатися як математичні моделі ЦП, тому що віддзеркалюють різними мовами один і той самий зв'язок вихідного сигналу з вхідними. Розглянемо способи перетворення між поширеними формами зображення функцій.

Від *словесного* або іншого способу задання логічної функції досить легко перейти до формального її опису. Для цього необхідно:

- 1) з'ясувати, скільки входів та виходів має пристрій;
- 2) встановити, які значення мають функції на кожному з виходів за всіх можливих комбінацій вхідних змінних;
- 3) скласти таблицю відповідності або безпосередньо ДДНФ функції.

Припустимо, логічна функція задана словесно таким чином: на виході пристрою встановлюється рівень лог. 1, якщо не менш, ніж на двох з трьох його входів діють рівні лог. 1. За та-

ким описом будуюмо таблицю відповідності (рис. 6.3, а), яка містить три змінні та одну функцію у, за одиничними значеннями якої безпосередньо з таблиці записуємо її ДДНФ (подано стрілками). Така функція реалізується *мажоритарним елементом*, загальне умовне позначення якого наведено на рис. 6.3, б (у прикладі $M=2$). Так само за нульовими значеннями функції легко перейти до її ДКНФ.

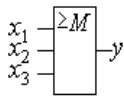
Якщо логічна функція задана *виразом довільної форми*, наприклад,

$$y = \overline{x_1 + x_2} + \overline{x_1 + x_3} + \overline{x_2 + x_3},$$

то для переходу до ДДНФ (або ДКНФ) спочатку можна скласти таблицю відповідності за методом прямої індукції (див. рис. 6.3, а): обчислюються логічні значення компонентів виразу для кожного кортежу, та, як підсумок, значення всієї функції у зводяться в колонку. Далі переходимо до ДДНФ як звичайно.

i	x_3	x_2	x_1	$\overline{x_1} + \overline{x_2}$	$\overline{x_1} + \overline{x_3}$	$\overline{x_2} + \overline{x_3}$	y
0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0
2	0	1	0	0	0	0	0
3	0	1	1	1	0	0	1
4	1	0	0	0	0	0	0
5	1	0	1	0	1	0	1
6	1	1	0	0	0	1	1
7	1	1	1	1	1	1	1

а



б

$y = x_1 x_2 x_3 + \overline{x_1} x_2 x_3 + x_1 \overline{x_2} x_3 + x_1 x_2 \overline{x_3}$

Рис. 6.3. Таблиця відповідності (а) та умовне позначення мажоритарного елемента (б)

4. Аналітичний перехід до досконалих форм. За іншим способом від вихідної формули переходимо до ДДНФ алгебрично:

а) позбуваючись інверсій над виразами (залишаючи їх, у разі потреби, тільки над окремими змінними), переходимо за законом двоїстості до ДНФ, тобто в нашому прикладі маємо:

$$y = x_1 x_2 + x_1 x_3 + x_2 x_3;$$

б) розгортаємо терми (компоненти) ДНФ таким чином,

щоб кожний з них містив усі змінні (за аксіомами $x \cdot 1 = x$, $1 = x + \bar{x}$):

$$y = x_1 x_2 \cdot 1 + x_1 \cdot 1 \cdot x_3 + 1 \cdot x_2 x_3 = x_1 x_2 (x_3 + \bar{x}_3) + x_1 (x_2 + \bar{x}_2) x_3 + (x_1 + \bar{x}_1) x_2 x_3 = \\ = x_1 x_2 x_3 + x_1 x_2 \bar{x}_3 + x_1 \bar{x}_2 x_3 + x_1 \bar{x}_2 \bar{x}_3 + \bar{x}_1 x_2 x_3 + \bar{x}_1 x_2 \bar{x}_3;$$

в) якщо виникають дублікації (повторення однакових термів), скорочуємо їх (за аксіомою: $x + x = x$) і здобуємо остаточний вираз ДДНФ:

$$y = x_1 x_2 x_3 + \bar{x}_1 x_2 x_3 + x_1 \bar{x}_2 x_3 + \bar{x}_1 \bar{x}_2 x_3;$$

г) за необхідністю відтворюємо таблицю відповідності у зворотному порядку: заступаючи літерали в мінтермах їх цифровими значеннями (змінну з інверсією замінюємо нулем, а без інверсії – одиницею, наприклад, $\bar{x}_3 x_2 x_1 \rightarrow 011_2 = 3_{10} = i$), визначаємо коди i вхідних кортежів, що відповідають значенням $y=1$, а на інших кортежах записуємо $y=0$.

Аналогічно перетворюємо функцію y до ДКНФ: спочатку позбуваємося інверсій над виразами і розгортаємо функцію до КНФ за розподільчим законом 8б (див. табл. 6.2):

$$y = x_1 x_2 + x_1 x_3 + x_2 x_3 = (x_1 + x_1 + x_2)(x_1 + x_1 + x_3)(x_1 + x_3 + x_2) \times \\ \times (x_1 + x_3 + x_3)(x_2 + x_1 + x_2)(x_2 + x_1 + x_3)(x_2 + x_3 + x_2)(x_2 + x_3 + x_3) = \\ = (x_1 + x_2 + x_3)(x_1 + x_2)(x_1 + x_3)(x_2 + x_3).$$

Відтак доповнюємо кожну диз'юнкцію змінними, яких бракує для утворення досконалої форми (за аксіомами $x + 0 = x$, $0 = x \bar{x}$):

$$A + B = A + B + 0 = A + B + C \bar{C} = (A + B) + C \bar{C} = (A + B + C)(A + B + \bar{C})$$

Після виключення дублікацій типу $x \cdot x = x$ у розглядуваному прикладі дістанемо ДКНФ

$$y = (x_1 + x_2 + x_3)(x_1 + x_2 + \bar{x}_3)(x_1 + \bar{x}_2 + x_3)(\bar{x}_1 + x_2 + x_3).$$

Зв'язок між ДКНФ і таблицею відповідності встановлюємо аналогічно як для ДДНФ: макстерми (вирази в дужках) визначають нульові набори функції, наприклад, $x_1 + x_2 + \bar{x}_3 \rightarrow \rightarrow 001_2 = 1_{10} = i$, а всі інші набори відповідають значенню функ-

ції $y=1$.

Таким чином, якщо функціонування пристрою задано логічним виразом, то виконуємо перетворення в загальному випадку за послідовністю: мішана форма – ДФ (КФ) – ДНФ (КНФ) – ДДНФ (ДКНФ). І, нарешті, від функції в ДДНФ (ДКНФ) легко перейти до її інверсії відповідно в ДКНФ (ДДНФ) взаємною заміною прямих та інверсних змінних і знаків диз'юнкції та кон'юнкції згідно з (6.1).

6.1.5. Співвідношення між логічними функціями

Алгебра логіки ізоморфна (взаємно відповідає) алгебрі множин відносно операцій об'єднання (диз'юнкції), перетину (кон'юнкції) та доповнення (інверсії). Це уможливорює використання діаграм Венна (діаграм Ейлера) для наочної ілюстрації властивостей алгебри логіки й розв'язання логічних рівнянь за допомогою алгебри множин.

1. Графічна ілюстрація операцій над логічними функціями. Розглядатимемо логічну функцію як множину $y=\{M_i\}$, елементами якої є мінтерми M_i , що відповідають значенню $y=1$ на певних вхідних кортежах i .

Якщо деяка функція є константою нуля $y=0$, що відповідає порожній множині $y=\emptyset$, то всю діаграму займає ділянка 0, а якщо вона є константою одиниці $y=1$, то вся діаграма збігається з ділянкою 1. Отже, співвідношення між частинами діаграми для довільної функції зображається як $\overline{y}=1\setminus y$ або $y=1\setminus \overline{y}$, тому що область визначення $U=y+\overline{y}\equiv 1$.

На діаграмі рис. 6.4, б зображено дві функції y_1 та y_2 , подані прямокутниками, а на спрощеній нижній діаграмі обидві функції подано рисками. Нумерація ділянок діаграми відповідає десятковим кодам, складеним з цих функцій, наприклад, ділянка, де перетинаються обидві функції, має код $i=y_2y_1=11_2=3_{10}$, а ділянка, де $y_2=1, y_1=0$ – номер $i=y_2\overline{y_1}=10_2=2_{10}$.

На цій діаграмі ділянки відповідають таким логічним операціям над функціями: 1, 3, 2 – диз'юнкція чи об'єднання (АБО) $D=y_1+y_2$; 0 – інверсія або доповнення об'єднання (АБО-

НІ) $\overline{D} = \overline{y_1 + y_2}$; 3 – кон'юнкція або перетин (І) $K = y_1 y_2$; 0, 1, 2 – доповнення перетину (І-НІ) $\overline{K} = \overline{y_1 y_2}$; 1 – різниця (заборона) $R_{12} = y_1 \setminus y_2$, яка утворюється вилученням з прямокутника y_1 тих одиниць, що збігаються з функцією y_2 ; 2 – аналогічна різниця $R_{21} = y_2 \setminus y_1$; 0, 2, 3 – імплікація $y_1 \rightarrow y_2 = R_{12}$; 0, 1, 3 – аналогічна імплікація $y_2 \rightarrow y_1 = \overline{R_{21}}$; 1, 2 – сума за модулем два (виключне АБО) $M = y_1 \oplus y_2$; 0, 3 – виключне АБО-НІ $\overline{M} = \overline{y_1 \oplus y_2}$.

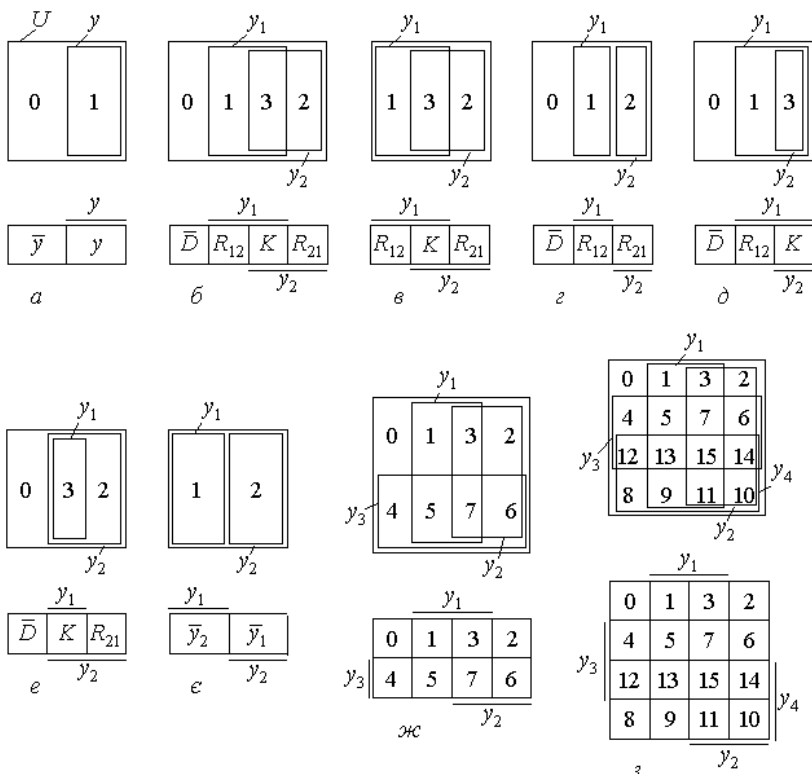


Рис. 6.4. Діаграми логічних функцій

Як видно з діаграми, підмножина мінтермів, яка утворює перетин функцій (ділянка 3), входить до складу підмножини об'є'нання (ділянки 1, 3, 2), що позначається символом включення $K \subset D$. Отже, завжди $KD = K$. Виражаючи подібним чином

операції одна через одну, дістанемо такі співвідношення:

$$\begin{aligned} D &= y_1 + y_2 = M + K; & K &= y_1 y_2 = D \oplus M \subset D; & KD &= K; \\ K + D &= D; & R_{12} &= y_1 \setminus y_2 = \overline{y_1 y_2} = y_1 \setminus K = y_1 \overline{K}; \\ R_{21} &= y_2 \setminus y_1 = \overline{y_1 y_2} = y_2 \setminus K = \overline{y_1} D; \\ M &= y_1 \oplus y_2 = y_1 \overline{y_2} + \overline{y_1} y_2 = R_{12} + R_{21} = D \overline{K}. \end{aligned} \quad (6.6)$$

2. Зв'язок між функціями і їх складниками. Якщо одну з двох функцій, наприклад, y_1 вважати відомою, то за допомогою операцій (6.6) можна виразити шукану функцію y_2 через відому або її окремі складники, що є, по суті, розв'язанням логічного рівняння відносно невідомої змінної. Безпосередньо з діаграми Венна (див. рис. 6.4, б) видно, що пряма функція y_2 утворюється логічним підсумовуванням ділянок 2, 3, а інверсна $\overline{y_2}$ – ділянок 0, 1. Отже, потрібно виразити ці ділянки через величини, які вважатимемо відомими, бо їх можна визначити з таблиці відповідності. Залежно від співвідношення між функціями розглянемо такі випадки.

а) У загальному випадку (див. рис. 6.4, б) з урахуванням (6.6) матимемо

$$y_2 = R_{21} + K = \overline{y_1} D + K = \overline{\overline{y_1} D + K} = \overline{\overline{y_1} \overline{K} + \overline{D}} = y_1 \oplus M. \quad (6.7)$$

Ці вирази мають універсальний характер, бо яку з порівнюваних функцій вважати відомою y_1 , а яку – шуканою y_2 , не має значення, аби-но їх назви в (6.6) і (6.7) були однаковими. Вирази (6.7) в окремих випадках співвідношення між функціями можна спростити.

б) У разі *суцільного покриття* області визначені обома функціями (рис. 6.4, в), тобто якщо $D = 1$, $K \neq 0$, отримаємо

$$y_2 = \overline{R}_{12} = \overline{y_1} + K. \quad (6.8)$$

в) Ознакою *неперетинних функцій* (рис. 6.4, з) є $K=0$, а також $R_{12}=y_1$, $R_{21}=y_2$, тому

$$y_2 = D \setminus y_1 = \overline{y_1} D. \quad (6.9)$$

Зауважимо, що при цьому співвідношення $y_1 \setminus y_2 = y_1$, $y_2 \setminus y_1 = y_2$ можуть виявитися корисними для спрощення логічних

виразів.

з) У випадку *включення* шуканої функції до складу відомої $y_2 \subset y_1$ (рис. 6.4, *д*), коли $D=y_1$, $K=y_2$, $R_{12}=0$, маємо:

$$y_2 = y_1 \setminus R_{12} = y_1 \overline{R_{12}}. \quad (6.10)$$

д) Навпаки, за *включення* відомої функції до складу шуканої $y_1 \subset y_2$ (рис. 6.4, *е*), коли $D=y_2$, $K=y_1$, $R_{12}=0$, маємо:

$$y_2 = y_1 + R_{21}. \quad (6.11)$$

е) І, нарешті, ознакою *взаємноінверсних* функцій (рис. 6.4, *є*) є $D=1$, $K=0$, отже,

$$y_2 = D \setminus y_1 = \overline{y_1}. \quad (6.12)$$

Діаграма для більшої кількості функцій утворюється таким чином, щоб фігура наступної функції поділяла всі ділянки попередньої діаграми на дві частини. Так, для трьох змінних (рис. 6.4, *ж*) прямокутник y_3 перетинає всі ділянки попередньої діаграми на рис. 6.4, *б*, а фігура y_4 (рис. 6.4, *з*) утворює діаграму для чотирьох змінних. На останніх двох діаграмах ділянки нумеруються кодами відповідно $i=y_3y_2y_1$ та $i=y_4y_3y_2y_1$. Нульова ділянка на цих діаграмах є доповненням диз'юнкції \overline{D} , а ділянка з кодом $i=7$ або $i=15$ відповідно є кон'юнкція, тобто перетин усіх функцій. Проте зміст кожної ділянки легко визначається її кодом. Так, п'я́а ділянка на цих діаграмах є виразом відповідно $y_3y_2y_1$ та $y_4y_3y_2y_1$, тобто немає потреби йменувати кожну з них своєю назвою.

Комбінуючи ділянки аналогічно діаграмі для двох функцій, можна будь-яку невідому функцію виразити через інші, вже відомі. Так, логічним підсумовуванням ділянок 4...7 на діаграмі трьох функцій дістанемо пряму функцію y_3 , а ділянок 0...3 – інверсну функцію $\overline{y_3}$. Проте процедуру розв'язання логічного рівняння з багатьма функціями доцільно спростити, позначивши шукану функцію як y_2 , а деяку композицію попередь синтезованих k функцій як y_1 , наприклад, $y_1 = y_{01} + y_{02} + \dots \oplus y_{0k}$ або $y_1 =$

$=y_0y_1y_2y_3 \dots$ ін. Тоді задача зводиться до діаграми на рис. 6.4, *б* та розв'язку за (6.6)...(6.12) за умови $y_1 \neq 1$. У іншому разі добирається інша композиція функцій y_{0i} , коли ця умова виконується.

На завершення звернімо ще раз увагу на нумерацію ділянок діаграм на рис. 6.4, *б, ж, з* – клітинок з кодами *i*. Сусідніми називатимемо клітинки, розташовані поруч по горизонталі й вертикалі (але не по діагоналі). При наведеній нумерації сусідні геометрично клітинки відповідають і сусіднім кодовим наборам (коди відрізняються лише одним розрядом). Тому сусідніми на рис. 6.4, *з* є й крайні по горизонталі або по вертикалі, а також кутові клітинки. Отже, діаграму можна уявити як розгортку торуватої поверхні: з'єднанням верхньої і нижньої її ліній утворюється циліндр, а з'єднанням ще лівої та правої ліній – тор.

Таким чином, за допомогою теоретико-множинного уявлення про логічні функції досягається не лише наочна ілюстрація логічних операцій і доведення співвідношень, але й полегшується встановлення зв'язку між ними, що дає змогу виражати функції одна через одну з метою спрощення їх реалізації. Приклади на застосування співвідношень між функціями наводяться в п. 6.3.

6.2. Мінімізація логічних функцій

Логічне проектування, особливо найпростіших пристроїв, можна здійснити шляхом тотожних перетворень логічних функцій, проте навіть у нескладних випадках виникає потреба у формалізації перетворень, аби дістати оптимальний варіант схеми, а досвід та інтуїцію використовувати лише на завершальних етапах проектування, зокрема, під час вибору найкращого варіанту для реалізації. Формалізована процедура переходу від ДДНФ (ДКНФ) до більш короткої форми та вибору з-поміж багатьох варіантів зображення функції найкоротшої з них називається *мінімізацією* логічної функції. Отже, мінімізація полягає в перетворенні від ДДНФ до мінімальної ДНФ (МДНФ) або від ДКНФ до мінімальної КНФ (МКНФ) функції.

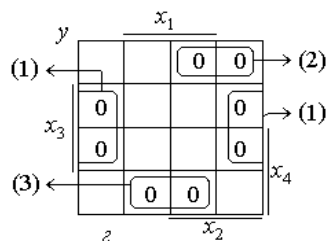
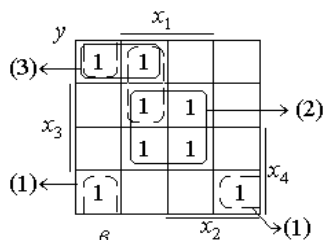
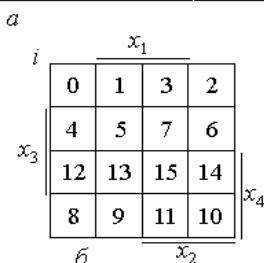
6.2.1. Графічний метод діаграм термів (Вайча - Карно)

1. Побудова діаграм. Серед формалізованих методів мінімізації логічних функцій найбільшого поширення набув метод за допомогою *діаграм термів*, де як математична модель використовується діаграма, клітинки якої є терми – члени виразу ДДНФ (мінтерми) або ДКНФ (макстерми). Такі діаграми мають поширену назву діаграм Вайча або карт Карно й будуються формально так само, як і діаграми Венна або діаграми Ейлера. Для стислості називатимемо їх діаграмами термів.

Для функції y чотирьох змінних (рис. 6.5, а) клітинки діаграми термів (рис. 6.5, б) нумеруються десятковими кодами вхідних кортежів $i = x_4 x_3 x_2 x_1$ так само, як і на діаграмі Венна (див. рис. 6.4, з). Типова нумерація клітинок полегшує заповнення власне діаграми функції y (рис. 6.5, в). Після позначення координат рисками, що відповідають одиничним значенням змінних, до діаграми переносять лише одиниці з колонки у таблиці відповідності згідно з кодами i вхідних кортежів, а до порожніх клітинок вважаються записаними нулі, або, навпаки, переносять лише нулі (рис. 6.5, з), тоді порожні клітинки відповідають значенням $y=1$.

Верхня половина діаграми на рис. 6.5, б відображає нумерацію

i	x_4	x_3	x_2	x_1	y
0	0	0	0	0	1
1	0	0	0	1	1
2	0	0	1	0	0
3	0	0	1	1	0
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	0
10	1	0	1	0	1
11	1	0	1	1	0
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	1



клітинок на діаграмі функції *трьох змінних*, а один верхній рядок – на діаграмі функції *двох змінних* аналогічно як і на діаграмах Венна (див. рис. 6.4, ж, б).

Діаграми функцій п'ятьох і більше змінних можна побудувати аналогічно подвоєнням попередніх діаграм (рис. 6.6, а, б, в). Так, для *п'ятьох змінних* координаті $x_5=0$ відповідає ліва відносно жирної осі піддіаграма (див. рис. 6.6, а), що повторює діаграму функції чотирьох змінних, а права піддіаграма (при $x_5=1$) нумерується дзеркально відносно осі (початок нумерацій виділено). Аналогічно за допомогою координати x_6 (рис. 6.6, в) утворюється діаграма функції *шістьох змінних*.

Діаграму, як і таблицю відповідності, можна побудувати й за будь-яким іншим способом задання функції, наприклад, за формулою

0	1	2	3	18	19	17	16
4	5	7	6	22	23	21	20
12	13	15	14	30	31	29	28
8	9	11	10	26	27	25	24

(1)	1	1		(3)	1	1	(2)		1	1	(1)
	*	*			1	1		1	1		
	1	*						1	1		
1	1							1	1		

0	1	2	3	18	19	17	16
4	5	7	6	22	23	21	20
12	13	15	14	30	31	29	28
8	9	11	10	26	27	25	24
40	41	43	42	58	59	57	56
44	45	47	46	62	63	61	60
36	37	39	38	54	55	53	52
32	33	35	34	50	51	49	48

ДДНФ або ДКНФ, бо кожній одиниці або нулю діаграми однозначно відповідає свій терм. Так, одиниці в лівому верхньому кутку діаграми на рис. 6.6, *б* відповідає мінтерм $x_1 x_2 x_3 x_4 x_5$, а нулю в правому верхньому кутку діаграми на рис. 6.5, *г* – макстерм $\overline{x_1 + x_2 + x_3 + x_4}$ згідно з правилами входження змінних до термів ДДНФ та ДКНФ.

2. Правила мінімізації. ДДНФ функції m змінних зображається як логічна сума всіх мінтермів рангу m , тобто добутків усіх m змінних або їх інверсій. Методи мінімізації ґрунтуються на наслідку склеювання (10а в табл. 6.2), який можна узагальнити таким чином:

$$Ax_i + \overline{Ax_i} = A, \quad (6.13)$$

де A – будь-який вираз, зокрема, добуток змінних. Отже, склеюватися можуть мінтерми, які відрізняються лише значенням однієї змінної: в одному з них $x_i=0$, а в іншому $x_i=1$. Саме так розташовано терми в діаграмах: сусідні клітинки (які знаходяться поруч по горизонталі й вертикалі або по краях діаграми) відрізняються лише однією змінною, тому дві одиниці в сусідніх клітинках склеюються графічно, утворюючи сполуку, що відповідає мінтермові нижчого рангу.

Графічна мінімізація за допомогою діаграм термів полягає в об'єднанні клітинок за *правилами*:

1) об'єднувати можна сусідні одиничні клітинки, утворюючи сполуки по $2^k=2, 4, 8, \dots$ одиниць, причому одна й та сама одиниця може входити до кількох сполук, а якщо якась одиниця не має сусідніх, вона утворює сполуку з однієї клітинки;

2) усі одиниці мають бути охоплені (покриті) сполуками таким чином, щоб, по-перше, кожна з них об'єднувала найбільшу кількість одиниць та, по-друге, діаграма містила найменшу кількість сполук;

3) будь-якій одиничній клітинці відповідає первісний мінтерм рангу m , а кожній сполуці з 2^k клітинок відповідає вкорочений мінтерм рангу $m-k$, що є добуток тільки тих змінних, які мають *стале значення для всієї сполуки*, причому змінна вхо-

дить до мінтерма без інверсії, якщо її значення є одиниця (є ризик проти всієї сполуки на діаграмі), та з інверсією – якщо нуль (ризика відсутня);

4) отже, мінімізована функція є логічною сумою всіх утворених мінтермів – сполук одиниць.

Здебільшого існує багато варіантів утворення сполук. Оптимізувати процедуру мінімізації доцільно таким чином.

1) Якщо є одиничні клітинки, які можна приєднати до інших лише одним способом, найкраще розпочати об'єднувати саме з таких клітинок. У прикладі на рис. 6.5, в більш, ніж одну сусідню мають усі одиничні клітинки (показано пунктиром), крім десятої (нумерацію клітинок подано на рис. 6.5, б). Тому утворюємо сполуку (1) приєднанням цієї клітинки до восьмої.

2) Після цього розглядаємо можливість утворити об'єднання з найбільшої кількості одиниць по 2^k . Так, на нашій діаграмі сполуку з 8 елементів утворити не можна, тому 4 одиниці об'єднуємо до квадрата (2).

3) Відтак із ще не охоплених одиниць утворюємо інші сполуки або приєднуємо вільні одиниці до вже утворених сполук так, щоб мінімальною кількістю сполук охопити (покрити) усі одиниці. У прикладі сполука (3) охоплює решту одиниць. Якщо варіантів рівноцінних сполучень кілька, занотовуємо їх для можливого використання на останньому етапі проектування.

4) І, нарешті, зчитуємо мінтерми з утворених сполук. Так, сполука (1), складена з двох первісних мінтермів четвертого рангу, утворює мінтерм третього рангу $\overline{x_1}x_2x_4$, до якого змінні x_1 , x_3 входять з інверсією тому, що проти всієї сполуки (1) немає їх ризиків, та змінна x_4 входить без інверсії тому, що проти всієї сполуки розташована її ризика, а змінна x_2 до мінтерма не входить через те, що її значення не є сталим протягом сполуки (проти клітинки 10 є ризик, а проти клітинки 8 ризика відсутня). Чотири одиниці сполуки (2) відповідають мінтерму другого рангу x_1x_3 , до якого входять змінні x_1 і x_3 без інверсії, бо проти всієї сполуки є ризики цих змінних, а змінні x_2 , x_4 до сполуки не входять, бо їх значення змінюються протягом сполуки.

Аналогічно зчитуємо мінтерм зі сполуки (3): $\overline{x_2} \overline{x_3} \overline{x_4}$. Підсумовуванням мінтермів дістаємо шукану МДНФ у вигляді

$$y = x_1 x_3 + x_2 x_3 x_4 + x_1 x_3 x_4.$$

Цю ж саму діаграму термів можна використати й для здобуття інверсної функції КНФ, якщо сполуки зчитувати як такі, до яких змінні входять з протилежним значенням (риска відповідає нульовому значенню аргумента, а її відсутність – одиничному). Так, безпосередньо з діаграми на рис. 6.5, *в* зчитуємо МКНФ інверсної функції

$$\overline{y} = (\overline{x_1} + \overline{x_3}) \cdot (x_2 + x_3 + x_4) \cdot (x_1 + x_3 + \overline{x_4}).$$

Очевидно, цей вираз можна дістати згідно з (6.1) і безпосередньо з формули МДНФ (6.14), якщо вона є.

Методика мінімізації в КНФ за допомогою діаграм термів не відрізняється від розглянутої з урахуванням лише особливостей (див. рис. 6.5, *з*): об'єднують до сполук клітинки з нулями та до складу макстермів змінну зчитують з інверсією, якщо проти сполуки є її риска і, навпаки, без інверсії, якщо риска відсутня. Тому МКНФ функції має вигляд:

$$y = (x_1 + \overline{x_3}) (\overline{x_2} + x_3 + x_4) (\overline{x_1} + x_3 + \overline{x_4}). \quad (6.15)$$

Інверсну функцію МДНФ також можна отримати з цієї самої діаграми, зчитуючи сполуки нулів так само, як сполуки одиниць:

$$\overline{y} = \overline{x_1} \overline{x_3} + \overline{x_2} \overline{x_3} \overline{x_4} + \overline{x_1} \overline{x_3} \overline{x_4},$$

або, якщо є, безпосередньо з МКНФ згідно з (6.1).

3. Особливості мінімізації зі збільшенням кількості змінних. На будь-якій діаграмі кожна клітинка має m сусідніх (m – кількість змінних), розташованих, наприклад, для функції чотирьох змінних ліворуч, праворуч, вгорі та знизу, включаючи крайні. Для функції п'ятьох змінних (див. рис. 6.6, *а*) крім чотирьох на своїй півдіаграмі додається ще п'ята сусідня клітинка на другій півдіаграмі, розташована симетрично відносно осі симетрії (жирна лінія), з кодом $i+16$, тобто сусідньою відносно, наприклад, п'ятої буде ще двадцять перша клітинка, а для функції

шістьох змінних (рис. 6.6, в) сусідньою з п'ятою є ще тридцять сьома клітинка. Взагалі, зі збільшенням кількості змінних на одиницю крім попередніх додається ще одна симетрична клітинка з кодом $i+2^{m-1}$ на сусідній півдіаграмі. Са ит і складює процедуру мінімізації функцій зі збільшенням кількості змінних.

Полегшити мінімізацію можна, впорядкувавши цей процес. Розглянемо його на прикладі діаграми п'яти змінних (рис. 6.6, б).

1) Утворюємо спочатку сполуки клітинок на одній півдіаграмі, наприклад, правій, де одиниць більше. У прикладі здобуваємо сполуки 1,2 з чотирьох елементів кожна (та ще беремо до уваги не помічену другу від правого краю колонку), а також сполуку 3 із двох одиниць.

2) Уявно перегинаючи діаграму по осі симетрії, перевіряємо, чи не накладаються утворені сполуки на такі самі групи клітинок лівої півдіаграми. Так, сполука 1 віддзеркалена на лівій півдіаграмі чотирма симетричними клітинками, які й приєднуємо до цієї сполуки тепер вже з 8 одиниць; квадрат 2 повністю не відображається на лівій півдіаграмі (показано зірочками) та колонка одиниць на ній розірвана, тому на правій півдіаграмі залишаємо сполуку 2 з чотирьох елементів, яка охоплює більше вільних одиниць, ніж колонка. Дві клітинки правої півдіаграми об'єднуються із симетричними на лівій, утворюючи сполуку 3 з чотирьох одиниць.

3) Переходимо до лівої півдіаграми й утворюємо так само сполуки з вільних одиниць, приєднуючи їх за змогою до інших сполук спочатку своєї, а відтак симетричної правої півдіаграми. У прикладі є одна вільна одиниця в клітинці $i=13$, яка приєднується до іншої ($i=9$) зі сполуки 1, утворюючи сполуку 4 з двох одиниць. Перевірка на симетрію з одиницями правої півдіаграми (обведено) збільшує сполуку 4 до чотирьох одиниць на двох півдіаграмах.

4) Зчитуючи сполуки по черзі, записуємо вираз для МДНФ функції:

$$y = \overline{x_2}x_3 + x_1x_3x_5 + \overline{x_1}x_2\overline{x_4} + x_1x_2x_4.$$

Принагідно зауважимо тут, що діаграму можна заповни-

ти за процедурою, зворотною зчитуванню термів під час мінімізації. Якщо функція задана в мішаній формі, спочатку перетворюємо її до нормальної форми (ДНФ або КНФ). Відтак визначаємо *групи клітинок*, що відповідають кожному термові і вносимо їх до діаграми. Наприклад, перший терм ДНФ (6.14) визначає квадрат одиниць (2) на діаграмі (див. рис. 6.5, в), приті кожного rischi змінних x_1, x_3 ; другому термові відповідає сполука (3), проти якої відсутні rischi змінних x_2, x_3, x_4 та ін. Так само від КНФ (6.15) легко перейти до діаграми (див. рис. 6.5, г) з урахуванням входження змінних до термів.

Таким чином, для мінімізації логічних функцій методом діаграм термів (Вайча-Карно) будуємо діаграму з 2^m клітинок, позначаємо її координати (нумерацію клітинок) таким чином, аби сусідні геометрично клітинки відповідали сусіднім кодовим наборам змінних та вносимо до діаграми за координатами одиниці (нулі) з таблиці відповідності. Відтак, об'єднуючи одиничні (нульові) сусідні клітинки за певними правилами, утворюємо сполуки, що відповідають мінтермам (макстермам) нижчого рангу та, зчитуючи сполуки, записуємо МДНФ (МКНФ) шуканої функції. Перевага методу полягає в його наочності під час ручного проектування, недолік – в ускладненні процедури мінімізації функцій зі збільшенням кількості змінних.

6.2.2. Аналітичні методи

1. Формалізований алгебричний метод. Здобуття найкращого або прийняттого варіанту логічної функції алгебрично, шляхом тотожних перетворень займає багато часу й не дає однозначності щодо оптимальності остаточного виразу. Аби уникнути цих недоліків, надамо перетворенням чітко формалізованої послідовності дій і розглянемо їх для наочності на прикладі логічної функції y , що подана таблицею відповідності (див. рис. 6.5, а) [12].

1) Зобразимо ДДНФ функції y у вигляді таблиці мінтермів M_4 (у прикладі – четвертого рангу) з кодами наборів змінних i , що відповідають одиничним значенням функції (рис. 6.7, а). Су-

ма цих мінтермів і становить ДДНФ функції. Згідно з (6.13) склеюватись можуть мінтерми, які відрізняються лише значенням однієї змінної, тому розташуємо їх у таблиці групами за кількістю інверсій змінних або, що те ж саме, нулів у їх кортежах і відокремимо групи рисками: у першій – чотири інверсії змінних, у другій – три тощо.

2) Внаслідок того, що склеюватись можуть лише мінтерми сусідніх груп, значно зменшується кількість потрібних перевірок на склеювання, які виконуємо послідовно для мінтермів з номерами 0 і 1, 0 і 8, відтак 1 і 5, 1 і 10 тощо. Якщо результат перевірки позитивний – утворюється мінтерм третього рангу, вносимо його до нової таблиці (рис. 6.7, б) мінтермів нижчого рангу M_3 з кодами i_3 , складеними з кодів склеєних мінтермів вищого рангу M_4 . Наприклад, перевірка на склеювання мінтермів з кодами $i=0$ та 1: $x_4 x_3 x_2 x_1 + x_4 x_3 x_2 x_1 = x_4 x_3 x_2$ дає мінтерм третього рангу з кодом $i_3=(0,1)$. Коди i можна вважати множинами, що вказують, з яких мінтермів M_4 складаються мінтерми M_3 . Такий алгоритм передбачає перевірку на склеювання між усіма можливими сполученнями мінтермів, що еквівалентно додаванню одного й того самого мінтерма M_4 кілька разів, проте на підставі аксіоми $x+x=x$ функція від цього не змінюється.

3) Далі перевіряємо на склеювання мінтерми M_3 , для чого поділяємо їх на групи з однаковими літералами (на рис. 6.7, в відокремлені рисками), бо склеюватись згідно з (6.13) можуть тільки мінтерми, складені з однакових змінних, одна з яких відрізняється входженням (без інверсії та з інверсією). Перевіряючи на склеювання всі можливі сполучення між мінтермами всередині кожної групи, одержуємо дві такі пари (відмічено квадратними дужками), які утворюють мінтерми меншого, тепер другого рангу M_2 , і вносимо їх до таблиці (рис. 6.7, з). У прикладі вони однакові (бо множини i_2 складені з однакових елементів), тому, усуваючи дублікацію, об'єднуємо їх в один мінтерм $x_3 x_1$ (показано дужкою).

Після другого склеювання функція у дорівнює сумі мінтермів M_2 (у нашому випадку він один) і M_3 , що не увійшли до складу M_2 , а також, якщо є, M_4 , що не склеїлись (у прикладі та-

4) Утворена ДНФ функції є правильною лише тоді, коли на всіх вхідних кортежах i , що відповідають первісним мінтермам ДДНФ M_4 , вона обертається на одиницю, а на всіх інших кортежах – на нуль. Проте, якщо вилучення з ДНФ якогось з компонентів, наприклад, одного з мінтермів M_3 , не змінює функцію на його кортежі, то він є зайвий. Перевіряючи на зайвину по черзі всі складники ДНФ та їх сполучення, можна дістати мінімізовану функцію. Але такі обчислення та порівняння варіантів займають багато часу, тому цей процес також потребує формалізації.

i_2	M_2	i	i_3				i_2
			(0,1)	(0,8)	(1,5)	(8,10)	(5,7,13,15)
(5,7,13,15)	$x_3 x_1$	0	+	+			
(5,13,7,15)	$x_3 x_1$	1	+		+		
		8		+		+	
ε		10				+	
		5			+		+
		7					+
		13					+
		15					+
		∂					

Для цього складаємо спочатку таблицю (рис. 6.7, *д*) від-

повідностей кортежів зі складниками ДНФ – множинами i_3, i_2 : деяким знаком, наприклад, плюс позначаємо, які з кортежів складають кожну з множин. Зокрема, до множини $i_3=(0,1)$ як елементи входять набори $i=0$ та $i=1$, тому на їх перетині ставимо знак плюс.

Далі виходимо з того, що всі кортежі i мають бути репрезентовані в ДНФ таким чином, аби кількість її складників – множин – була мінімальною. Передусім, природно, залишаємо мінтерми найнижчого рангу (екстремалі) M_2 , тобто множини i_2 (на рис. 6.7, δ обведено), бо вони вбирають найбільшу кількість мінтермів M_4 – кортежів i . Відтак відокремлюємо в таблиці ще не охоплені (не покриті) кортежі i та множини, до складу яких вони входять, перебираємо можливі сполучення множин i_3 , що охоплюють усі кортежі, та залишаємо тільки мінімальні з цих сполучень. Так, сполучення по два $(0,1)$ і $(0,8)$, $(0,1)$ і $(1,5)$ не забезпечують покриття, а $(0,1)$ і $(8,10)$ покривають усі кортежі (у таблиці обведено). Якщо немає таких сполучень по два, переходимо до сполучень по три, а якщо рівноцінних сполучень кілька, занотовуємо їх також як варіанти розв'язку задачі.

5) І, нарешті, як підсумок, записуємо логічний вираз (або вирази, якщо є рівноцінні варіанти) МДНФ з дібраних множин, розшифровуючи їх за кодами з відповідних таблиць M_2 та

i	M_4			
	x_4	x_3	x_2	x_1
0	0	0	0	0
1	0	0	0	1
8	1	0	0	0
5	0	1	0	1
10	1	0	1	0
7	0	1	1	1
13	1	1	0	1
15	1	1	1	1

a

i_3	M_3			
	x_4	x_3	x_2	x_1
(0,1)	0	0	0	X
(0,8)	X	0	0	0
(1,5)	0	X	0	1
(8,10)	1	0	X	0
(5,7)	0	1	X	1
(5,13)	X	1	0	1
(7,15)	X	1	1	1
(13,15)	1	1	X	1

б

i_2	M_2			
	x_4	x_3	x_2	x_1
(5,7,13,15)	X	1	X	1
(5,13,7,15)	X	1	X	1

в

Рис. 6.8. Використання алгоритму Квайна – Мак-Класкі

M_3 :

$$y = x_1 x_3 + \overline{x_2} \overline{x_3} x_4 + x_1 x_3 x_4.$$

Природно, ця формула збігається з (6.14) і є значно простіша за вихідну ДДНФ.

2. Алгоритм Квайна – Мак-Класкі. За своєю сутністю розглянутий метод не відрізняється від аналітично-табличного методу – алгоритму Квайна – Мак-Класкі. Процедура синтезу за цим методом ще більш формалізована шляхом заміни мінтермів M_4, M_3, M_2 їх цифровими кодами (рис. 6.8, *а, б, в*): у виразах змінні x_i замінено відповідно одиницею або нулем. Якщо під час склеювання утворюється мінтерм нижчого рангу, відсутню змінну (прогалина на рис. 6.7, *б, в*) замінюємо позначкою X . Це спрощує запис і уможливорює перетворення на ЕОМ, бо зі збільшенням кількості змінних різко зростає кількість мінтермів, що підлягають перевірці на склеювання. Останній етап мінімізації здійснюється за таблицею, поданою на рис. 6.7, *д* [12].

Таким чином, різновиди аналітичних методів мінімізації логічної функції, зокрема, за алгоритмом Квайна – Мак-Класкі полягають у перевірці мінтермів на склеювання та добірї найменшої кількості здобутих скорочених мінтермів (імплікант), які покривають усі первісні мінтерми. Цим здійснюється перетворення від ДДНФ до МДНФ. Метод застосовується, головним чином, для мінімізації функцій з багатьма (не менш, ніж п'ятьма-шістьма) аргументами, особливо на ЕОМ.

6.3. Основи схемної реалізації логічних функцій

6.3.1. Реалізація в поширених базисах

1. Булів базис. Якщо функція зображена в ДНФ або в КНФ, то вона містить три елементарні логічні операції над змінними (не над виразами), що складають булів базис. Схемно такі функції реалізуються сполученням входів і виходів логічних елементів у послідовності: НІ, І, АБО відповідно до формул у ДНФ або в послідовності: НІ, АБО, І за формулами в КНФ. Так,

згідно з виразами ДДНФ (6.2) та ДКНФ (6.4) можна побудувати в булевому базисі елемент виняткове АБО (рис. 6.9, а, б).

Проте зображення схем у булевому базисі, здебільшого, має розглядатися лише як основа для їх порівняння і подальшого перетворення до *структурної* логічної функції. З метою вкорочення схем, використання вільних логічних елементів усередині корпусів частково задіяних мікросхем, зменшення кількості їх типономіналів, зокрема, у потоковому виробництві, а також використання основних і, часто, кращих за параметрами логічних елементів певної серії, доводиться перетворювати схеми або їх фрагменти від одного базису до іншого. Для цього розглянемо принципи побудови схем у більш поширених базисах.

2. Базис І-НІ. Реалізація елементів НІ, І, АБО, АБО-НІ (рис. 6.10, а, б, в, г) у базисі І-НІ впливає безпосередньо з співвідношень алгебри логіки. Наприклад, за законом двоїстості дістаємо в цьому базисі функцію АБО (див. рис. 6.10, в):

$$\overline{x_1 + x_2} = \overline{x_1} \cdot \overline{x_2}.$$

Складніші, комбіновані функції, зображені в булевому базисі, можна здобути шляхом сполучення елементів НІ, І, АБО, реалізованих у базисі І-НІ, та наступним вилученням послідовно ввімкнених інверторів (якщо вони є) за аксіомою подвійного заперечення. Так, з'єднанням двох елементів І з входами елемента АБО (рис. 6.10, д) та вилученням двох пар послідовно ввімкнених елементів НІ отримуємо схему, еквівалентну елементу І-АБО (рис. 6.10, е), а додаванням на виході інвертора – схему І-АБО-НІ (рис. 6.10, є). Використовуючи елементи НІ та І-АБО, переходимо так само від схеми виняткове АБО в булевому базисі (див. рис.

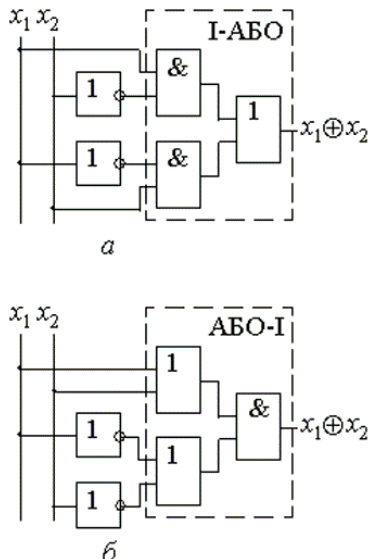


Рис. 6.9. Реалізації логічного елемента виняткове АБО

6.9, а) до схеми в базисі І-НІ (рис. 6.10, ж).

Проте реалізацію в базисі І-НІ зазвичай виконують не перетворенням схем, а алгебрично, за формулою:

$$y = A + B = \overline{\overline{A + B}} = \overline{\overline{A} \cdot \overline{B}}, \quad (6.16)$$

де A, B – кон'юнкції літералів (змінних або їх заперечень). Тобто для такого перетворення необхідно: а) записати функцію в ДНФ (МДНФ або, якщо не підлягає мінімізації, ДДНФ), б) поставити над виразом два знаки інверсії та в) за законом двоїстості перейти від суми до добутку змінних або їх кон'юнкцій. Наприклад, для функції виняткове АБО, виходячи з ДДНФ (6.2), дістаємо структурну функцію $y = \overline{x_1 x_2} + \overline{x_1 x_2} = \overline{\overline{\overline{x_1 x_2} + \overline{x_1 x_2}}} = \overline{\overline{\overline{x_1 x_2} \cdot \overline{x_1 x_2}}}$, що відповідає рис. 6.10, ж.

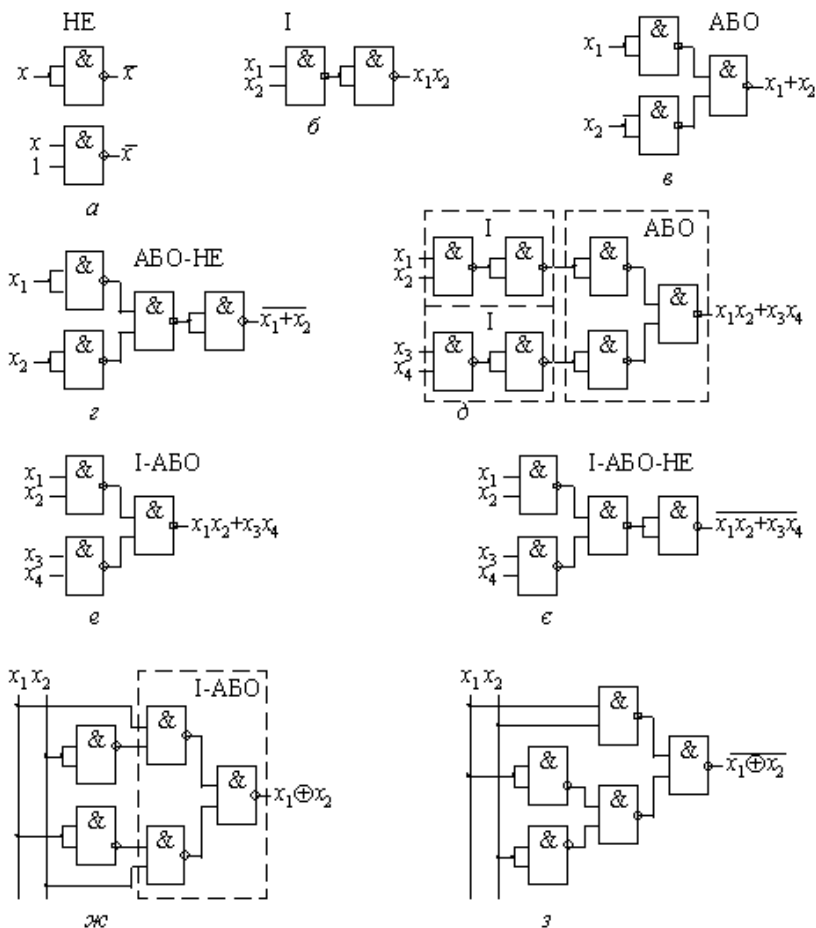


Рис. 6.10. Схемна реалізація логічних функцій в базисі І-НІ

Якщо функцію задано в КНФ, то спочатку доцільно перейти до ДНФ, а відтак – до базису І-НІ. Наприклад, вираз (6.4) за допомогою (6.1) перетворюємо таким чином:

$$y = (x_1 + x_2)(\overline{x_1} + \overline{x_2}), \quad \overline{y} = \overline{x_1 x_2 + x_1 x_2} = \overline{x_1 x_2} \cdot \overline{x_1 x_2}$$

і реалізуємо елемент виняткове АБО-НІ (рис. 6.10, з).

3. Базис АБО-НІ. Реалізація логічних елементів НІ, АБО, І, І-НІ (рис. 6.11, а, б, в, г) та перетворення схем від булевого ба-

зису до базису АБО-НІ шляхом заміни елементів (див. рис. 6.9, б) їх еквівалентами в базисі АБО-НІ з усуненням пар послідовно ввімкнених інверторів, якщо вони є (рис. 6.11, д, е, є, ж), повністю дуальні перетворенням у базисі І-НІ.

Алгебрично для переходу до базису АБО-НІ перетворення виконують за формулою:

$$y = A \cdot B = \overline{\overline{A \cdot B}} = \overline{\overline{A} + \overline{B}} \quad (6.17)$$

(тут A, B – диз'юнкції літералів), тобто в такому порядку: а) записують функцію в КНФ (мінімальну, якщо вона є, інакше ДКНФ), б) позначають над виразом два знаки інверсії та в) переходять від добутку до логічної суми за законом двоїстості.

Від функції, заданої в ДНФ, попередньо згідно з (6.1) переходять до КНФ, а відтак – до базису АБО-НІ. Наприклад, перетворенням (6.2)

$$y = \overline{x_1 x_2} + \overline{x_1 x_2};$$

$$\overline{y} = (\overline{x_1} + \overline{x_2})(\overline{x_1} + \overline{x_2}) = \overline{\overline{\overline{x_1} + \overline{x_2}}} = \overline{\overline{\overline{x_1} + \overline{x_2}}}$$

реалізується елемент виняткове АБО-НІ (рис. 6.11, з).

4. Базис І-АБО-НІ. Легко показати за законами двоїстості

$$\overline{\overline{x_1 x_2} + \overline{x_3 x_4}} = (\overline{x_1} + \overline{x_2})(\overline{x_3} + \overline{x_4})$$

реалізацію функції АБО-І в базисі І-АБО-НІ (рис. 6.12, а). Взагалі перетворення схем виконують алгебрично за формулами

$$\overline{y} = \overline{AB + CE}; \quad y = \overline{\overline{AB} + \overline{CE}} \quad (6.18)$$

тобто в послідовності: а) записують вираз інверсної функції в ДНФ та б) за допомогою інверсії над обома частинами виразу переходять до структурної функції в базисі І-АБО-НІ.

Наприклад, з ДНФ (6.3)

$$\overline{y} = \overline{x_1 x_2} + \overline{x_1 x_2}; \quad y = \overline{\overline{\overline{x_1 x_2} + \overline{x_1 x_2}}}$$

дістаємо структурну функцію, зручну для побудови елемента виняткове АБО (рис. 6.12, б).

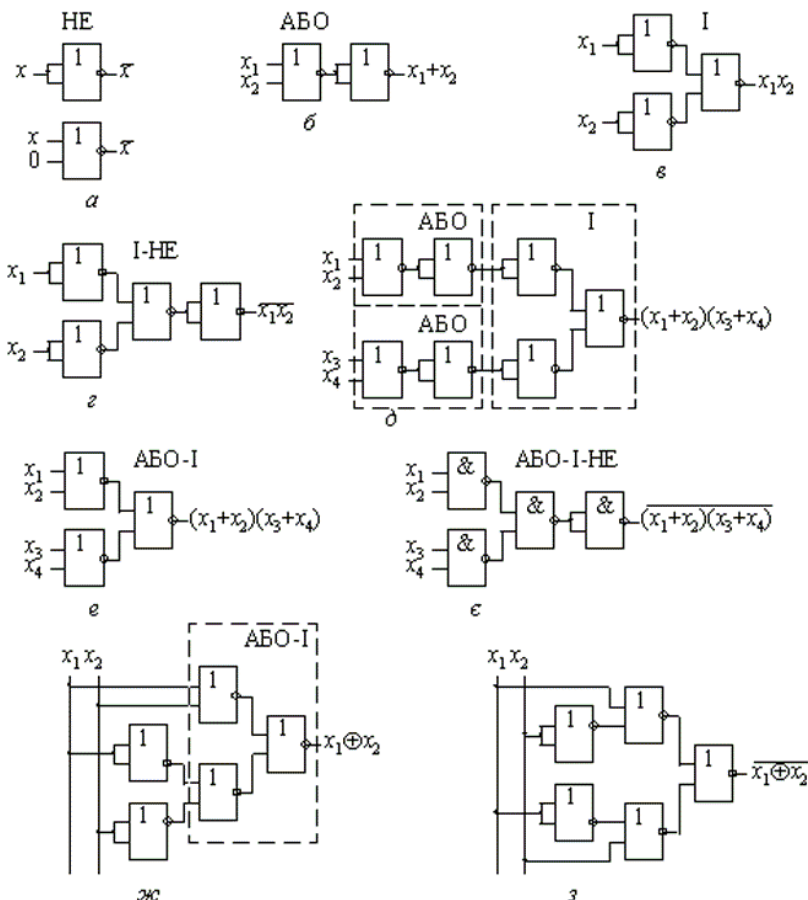


Рис. 6.11. Схемна реалізація логічних функцій в базисі АБО-ІІ

Таким чином, вихідною для подальших перетворень з метою спрощення схеми та побудови її у потрібному базисі є мінімізована функція ДНФ або КНФ, що має зображення в булевому базисі, тобто на елементах ІІ, І, АБО. У заданому базисі функцію реалізують шляхом її алгебричного перетворення за певними правилами. При цьому до базисів І-ІІ та І-АБО-ІІ простіше перетворюються функції, зображені в ДНФ, а до базису АБО-ІІ – в КНФ.

6.3.2. Способи спрощення логічних схем

1. Проблема схемної мінімізації. Розглянута мінімізація логічної функції в МДНФ та МКНФ відображає її в булевому базисі, який є малоприсадаблений щодо технічної реалізації. Серії сучасних інтегрованих мікросхем містять різні комплекти стандартних логічних елементів, у тому числі багатовходових, використання яких дозволяє *мінімізувати схеми*. Проте в загальному вигляді проблема схемної мінімізації в різних функційно повних системах на цей час не розв'язана.

У багатьох випадках мінімізація функції передуватиме переходу від булевого до інших базисів та подальшим перетворенням з метою спрощення структурної функції.

Аби полегшити подібні перетворення, далі розглядаються окремі прийоми схемної мінімізації, тобто способи зменшення складності логічної схеми. Як показник

її складності зазвичай використовується потрібна кількість корпусів q_k однотипних ІС. На стадії логічного проектування відносно простих схем користуються показником складності структурної функції у вигляді суми $q = q_e + q_v$ потрібної кількості логічних елементів q_e та загальної кількості їх входів q_v (іноді складність зручніше оцінювати потрібною загальною кількістю зовнішніх виводів інтегрованих мікросхем $q_{з.в.}$). Проте наочним для порівняння схем і зручним під час їх технічної реалізації на логічних елементах (на вентильному рівні) є показник у вигляді зазначення через риску кількості елементів на загальну кількість їх входів $q = q_e / q_v$, яким і користуватимемося далі.

Складність легко оцінити безпосередньо з виразу підрахування кількості логічних операцій та змінних (літер), що

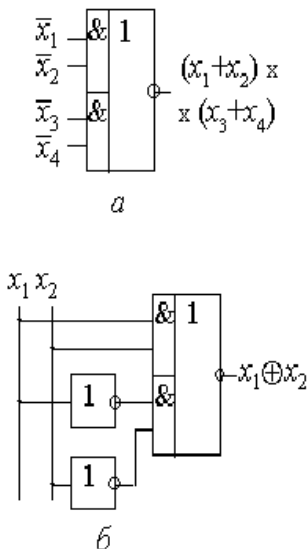


Рис. 6.12. Використання І-АБО-НІ

входять до кожної з них. Так, у виразі (6.2) маємо два елементи І та один АБО по два входи кожний, а також два інвертори по одному входу, тобто разом $q=5/8$ (див. рис. 6.9, а). У виразах, перетворених до базисів І-НІ та АБО-НІ, досить підрахувати кількість знаків інверсії (елементи) та змінних під кожним з них (входи).

2. Редукція в диз'юнктивній формі. Якщо об'єднанню до сполук з більшої кількості одиниць на діаграмі термів заважають окремі нулі, функцію можна спростити, розглядаючи спочатку ці нулі як одиниці, а відтак скоригувати її вилученням з множини одиниць зайві нульові клітинки. Систематизовану процедуру такого спрощення називають редукцією. У загальному вигляді під час мінімізації в ДНФ редукцію можна виконати доповненням заданої функції у неперетинною з нею допоміжною функцією y_1 таким чином, аби об'єднання D цих функцій та сама допоміжна функція y_1 були якомога простішими. Тоді, на підставі (6.9) шукану функцію дістаємо з виразу

$$y = D \setminus y_1 = \overline{D y_1}. \quad (6.19)$$

Так, мінімізацією звичайним чином функції, заданої діаграмою 1 на рис. 6.13, а

$$y = \overline{x_1 x_2} + \overline{x_2 x_4} + \overline{x_1 x_3 x_4} = \overline{x_1 x_2} \cdot \overline{x_2 x_4} \cdot \overline{x_1 x_3 x_4},$$

маємо реалізацію її в базисі І-НІ складністю $q=7/13$. Редукцію виконуємо у такий спосіб. Введенням допоміжної функції $y_1 = x_1 x_2$ (зазначена зірочками) мінімізуємо утворене об'єднання (на діаграмі обведено)

$$D = x_1 + \overline{x_2 x_4} + \overline{x_3 x_4} = x_1 + \overline{x_2 x_3 x_4}$$

і за (6.19) у підсумку отримуємо структурну функцію

$$y = \overline{D y_1} = (\overline{x_1 + \overline{x_2 x_3 x_4}}) \overline{x_1 x_2} = \overline{x_1 x_1 x_2} \cdot \overline{x_1 x_2 x_2 x_3 x_4},$$

яка реалізується мінімальною схемою в базисі І-НІ складністю $q=5/11$ (рис. 6.13, б).

З метою спрощення допоміжної функції до неї можна додати інші нульові клітинки (якщо вони є), які не входять до об'єднання. При цьому корекція за виразом (6.19) не змінює функцію, бо згідно з операцією заборони $0 \setminus 1 = 0$.

Якщо в нашому прикладі утворити об'єднання, обведене на діаграмі (2) $D=x_1+x_4$, то для вилучення зайвих клітинок, позначених зірочками, до допоміжної функції (пунктир) додаємо ще шосту клітинку: $y_1=x_1x_2+x_2x_3$. У підсумку маємо шукану функцію

$$y = \overline{Dy_1} = (x_1 + x_4) \overline{x_1x_2 + x_2x_3} = \overline{x_1x_1x_2 + x_2x_3 \cdot x_1x_2 + x_2x_3x_4},$$

яка реалізується мінімальною схемою з використанням елемента І-АБО-НІ складністю $q=4/10$ (рис. 6.13, в).

Цілком зрозуміло, що редукцію можна застосувати і не до функції в цілому, а лише до її фрагменту, якщо така процедура приводить до мінімальної форми. Тоді, під час коригування фрагменту за (6.19) до допоміжної функції з метою її спрощення можна долучити одиничні клітинки, що перетинаються з рештою функції. Це означає, що мінтерм, охоплений однією сполукою одиниць, не обов'язково повторювати в інших сполуках.

У прикладі на діаграмі (3) частину функції x_4 (верхня половина діаграми) не має потреби коригувати, а з решти одиниць та нульової клітинки, поміченої зірочкою, утворюємо об'єднання $D=x_1+x_3$ (обведено суцільною лінією). Для вилучення нульової клітинки до допоміжної функції $y_1=x_1x_2x_3$ додаємо ще сьому клітинку, бо вона вже репрезентована у функції частиною x_4 . Отже, маємо структурну функцію

$$y = \overline{x_4 + Dy_1} = (x_1 + x_3) \overline{x_1x_2x_3} + \overline{x_4} = \overline{x_1x_1x_2x_3 \cdot x_3x_1x_2x_3x_4}, \quad (6.20)$$

для реалізації в базисі І-НІ за мінімальною схемою (рис. 6.13, г) складністю $q=4/10$ (проти 7/13 або 6/12 у варіантах без редукції).

3. Редукція в кон'юнктивній формі. Враховуючи дуальність алгебри логіки щодо логічних функцій І та АБО, аналогічну редукцію можна виконати і під час мінімізації в КНФ. Але тепер до нульових клітинок функції y з метою спрощення їх сполук на діаграмі долучаємо ще деякі одиничні клітинки, утворюючи допоміжну функцію y_1 , якщо доповнення об'єднання \overline{D} цих функцій (тобто сполучення нульових і ще

приєднаних одиничних клітинок) і нова функція y_1 виявляються простими. Для здобуття шуканої функції необхідно в утвореному об'єднанні відновити одиничні клітинки, тобто виконати операцію імплікації

$$y = \overline{D} \leftarrow y_1 = \overline{D} + \overline{y_1}. \quad (6.21)$$

Так, МКНФ функції, що задана діаграмою (4) на рис. 6.13, а, перетворена до базису АБО-НІ, відповідає складності $q=7/13$:

$$y = (x_1 + x_2)(\overline{x_2} + x_4)(\overline{x_1} + \overline{x_3} + x_4) = \overline{\overline{x_1 + x_2 + x_2 + x_4 + x_1 + x_3 + x_4}}.$$

Для редукції спочатку з одиничних клітинок (помічені зірочками) утворюємо допоміжну функцію, яку (вважаючи ці клітинки нульовими) зображаємо в МКНФ: $y_1 = x_1 + x_2$. Відтак долученням цих клітинок до нульових утворюємо доповнення об'єднання (обведено суцільною лінією) і також мінімізуємо його в МКНФ:

$$\overline{D} = x_1(\overline{x_2} + x_4)(\overline{x_3} + x_4) = x_1(\overline{x_2 x_3} + x_4) = x_1(\overline{x_2 + x_3} + x_4).$$

Нарешті, за (6.21) дістаємо спрощений вираз шуканої функції

$$y = \overline{D} + \overline{y_1} = x_1(\overline{x_2 + x_3} + x_4) + \overline{x_1 + x_2}$$

та користуючись розподільчим законом $ab+c=(a+c)(b+c)$ переходимо до базису АБО-НІ

$$\begin{aligned} y &= (\overline{x_1 + x_1 + x_2})(\overline{x_2 + x_3 + x_4 + x_1 + x_2}) = \\ &= \overline{\overline{x_1 + x_1 + x_2 + x_2 + x_3 + x_1 + x_2 + x_4}} \end{aligned}$$

зі складністю $q=5/11$ (відповідає схемі на рис. 6.13, б, якщо елементи І-НІ замінити на АБО-НІ).

4. Алгебричне виконання редукції. Наведена в п. 2, 3 графічна процедура редукції за допомогою діаграм термів є наочною і практично не потребує тотожних перетворень. В простих випадках редукція виконується алгебрично.

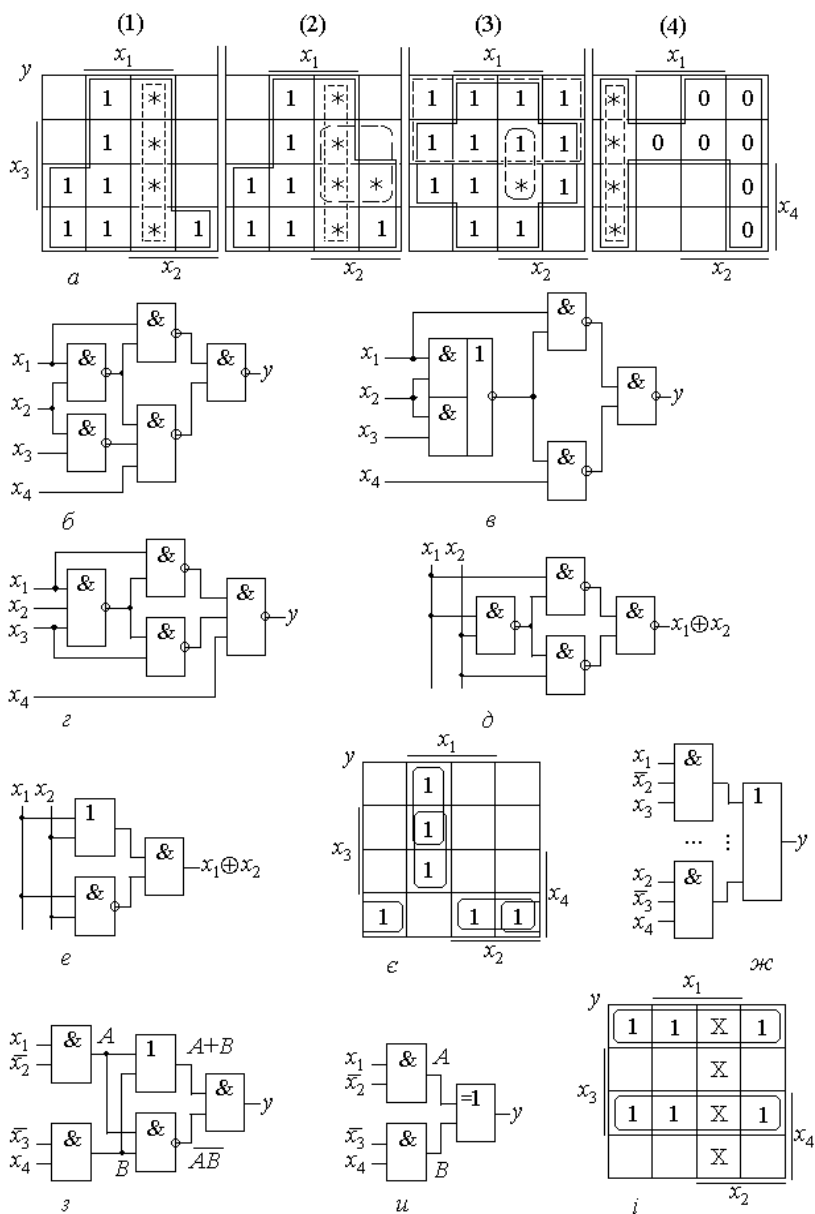


Рис. 6.13. Редукція в різних формах

Інколи перетворення спрощуються шляхом мінімізації в МКНФ функції для реалізації в базисі І-НІ та в МДНФ – для реалізації в базисі АБО-НІ. Наприклад, об'єднанням нулів на діаграмі 3 рис. 6.13, а, отримуємо МКНФ і за розподільчим законом (8б у табл. 6.2) зводимо до виразу

$$y = (x_1 + x_3 + \overline{x_4})(\overline{x_1} + \overline{x_2} + \overline{x_3} + \overline{x_4}) = \\ = (x_1 + x_3)(\overline{x_1} + \overline{x_2} + \overline{x_3}) + \overline{x_4} = (x_1 + x_3)\overline{x_1 x_2 x_3} + \overline{x_4},$$

від якого легко перейти до (6.20).

Алгебрично редукцію виконувати доцільно у випадку різного входження *однакових* змінних (без інверсії та з інверсією) до кількох термів безпосередньо за формулами в диз'юнктивній та кон'юнктивній формах відповідно:

$$A\overline{B} = A\overline{A} + A\overline{B} = A(\overline{A} + \overline{B}) = A\overline{A\overline{B}}; \quad (6.22)$$

$$A + \overline{B} = (A + \overline{A})(A + \overline{B}) = A + \overline{A\overline{B}} = A + \overline{A} + \overline{B}; \quad (6.23)$$

застосовуючи двічі перетворення за цими формулами, наприклад, до (6.2), (6.5):

$$y = \overline{x_1 x_2} + \overline{x_1 x_2} = \overline{x_1 x_1 x_2} + \overline{x_2 x_1 x_2} = \overline{x_1 x_1 x_2 \cdot x_2 x_1 x_2}; \quad (6.24)$$

$$\overline{y} = (\overline{x_1} + \overline{x_2})(\overline{x_1} + \overline{x_2}) = (\overline{x_1} + \overline{x_1} + \overline{x_2})(\overline{x_2} + \overline{x_1} + \overline{x_2}) = \\ = \overline{x_1} + \overline{x_1} + \overline{x_2} + \overline{x_2} + \overline{x_1} + \overline{x_2}, \quad (6.25)$$

дістанемо мінімальні схеми реалізації функцій виняткове АБО та виняткове АБО-НІ відповідно в базисах І-НІ та АБО-НІ складністю 4/8. На рис. 6.13 д наведено одну з двох цих дуальних схем.

Узагальнюючи, можна дійти висновку, що процедура редукції відображається на схемі додатковим рівнем сполучення логічних елементів. З утворених спрощених сполук діаграми зайві клітинки (разом з приєднаними, за змогою, сусідніми) вилучаються за допомогою тих логічних елементів вхідного рівня, виходи яких є *спільними* для входів кількох елементів наступного схемного рівня. Це добре видно з наведених прикладів на рис. 6.13, б, в, г, д.

5. Реалізація в мішаній формі. З огляду на те, що сучас-

ні серії інтегрованих мікросхем містять велику номенклатуру типоміналів, для спрощення схем широко використовують різні логічні елементи, тобто практично структурну функцію часто реалізують у мішаній формі.

Передусім, спрощення виконують зменшенням кількості інверсій, зокрема, без переходу до іншого базису. Так, формули (6.3), (6.4) перетворюють у вигляді

$$\overline{y} = \overline{x_1 x_2} + \overline{x_1 x_2}; \quad y = \overline{\overline{x_1 x_2} + \overline{x_1 + x_2}};$$

$$y = (x_1 + x_2) \overline{(x_1 + x_2)} = (x_1 + x_2) \overline{x_1 x_2}$$

і реалізують без додаткових інверторів як для прикладу на рис. 6.13, *е* подано схему для другої з цих мінімальних форм.

У мішаній формі з використанням складніших елементів, що виконують різні логічні функції, можна значно спростити схему. Прикладами є вже розглянута реалізація (6.3) в базисі І-АБО-НІ (див. рис. 6.12, *б*), а також відокремлення елемента виняткове АБО, наведене в п. 6.

6. Каскадна реалізація термів. Сутність такого спрощення полягає в тому, що в первісному по мінімізації виразі відокремлюють спільні для кількох термів частини винесенням їх за дужки, позначають деяким символом і використовують кілька разів для реалізації функції, внаслідок чого схема в цілому вкорочується.

Розглянемо такий прийом на прикладі функції, заданої таблицею термів на рис. 6.13, *є*. За сполуками одиниць дістаємо МДНФ

$$y = \overline{x_1 x_2 x_3} + \overline{x_1 x_2 x_4} + \overline{x_1 x_3 x_4} + \overline{x_2 x_3 x_4},$$

що реалізується схемою, фрагмент якої наведено на рис. 6.13, *ж*, складністю $q=9/20$ з урахуванням вхідних інверторів, які тут і далі не показані. Спрощення виконуємо таким чином. Позначаємо спільні частини $A = \overline{x_1 x_2}$, $B = \overline{x_3 x_4}$, виносимо їх за дужки та перетворюємо МДНФ з урахуванням (6.22)

$$y = A \overline{x_3} + A \overline{x_4} + \overline{x_1} B + x_2 B = A(\overline{x_3} + \overline{x_4}) + B(\overline{x_1} + x_2) = \overline{\overline{A x_3 x_4}} + \overline{\overline{B x_2 x_1}} = \overline{A \overline{B}} + \overline{B \overline{A}} = \overline{A \overline{B}} + \overline{B \overline{A B}} = (A + B) \overline{A B}.$$

Отже, складність схеми зменшується до $q=7/12$ (рис. 6.13, з). Якщо доповнити сполуки одиниць долученням дев'ятої клітинки, то такого самого результату можна дійти коротше, процедурою редукції.

Ще більше спрощується схема відокремленням елементу виняткове АБО

$$y = \overline{AB} + \overline{BA} = A \oplus B,$$

складність її становить $q=5/8$ (рис. 6.13, и).

7. Частково визначені функції. Якщо деякі вхідні кортежі за умов роботи пристрою не виникають, то функція є *частково визначеною*. Наприклад, у двійково-десяткових кодах (див. табл. 5.6, 5.8) набори змінних при $X_{10}>9$ є забороненими, тобто функція на цих наборах лишається невизначеною, що можна тлумачити як байдужі (факультативні) її значення $X=0$ або 1, бо вони не зустрічаються під час правильної роботи пристрою.

Хай функція чотирьох змінних набуває одиничного значення на кортежах: $i=0, 1, 2, 12, 13, 14$, крім того, забороненими є кортежі $i=3, 7, 11, 15$ та нульовими всі інші. На діаграмі термів (рис. 6.13, і) факультативні значення функції помічено позначкою X . З метою спрощення під час мінімізації функції невизначені умови *довизначають* таким чином, аби утворювались найбільші сполуки з одиниць, а далі мінімізують як звичайно. У прикладі в двох клітинках, які доцільно приєднати до одиниць, вважаємо $X=1$, а в двох інших – $X=0$. Зчитування по утворенні сполук дає спрощену МДНФ

$$y = x_3x_4 + \overline{x_3x_4} = \overline{x_3} \oplus x_4,$$

що є функцією виняткове АБО-НЕ. Не важко переконатись, що той самий результат отримаємо об'єднанням на діаграмі нулів і аналогічним довизначенням клітинок X .

Отже, чим більше існує заборонених кортежів, на яких функція є невизначеною, тим більше є можливості для її спрощення.

Таким чином, після мінімізації логічну функцію доцільно додатково спростити для реалізації її з урахуванням потрібного елементного базису. Для цього застосовують пере-

творення по зменшенню кількості інверсій в термах, переходу до структурної функції в мішаній формі, відокремленню спільних фрагментів у кількох термах з метою їх каскадної реалізації, використанню більш складних елементів, зокрема, типових пристроїв більшого ступеня інтеграції.

2.3.3. Спрощення логічних схем з багатьма виходами

Окрема реалізація функцій для кожного з виходів схем з багатьма виходами, здебільшого, спричиняє нераціональні витрати елементів. З метою оптимальної побудови схеми в цілому вдаються до *спільної реалізації* сукупності функцій. Для цього логічні функції перетворюють таким чином, аби вони містили спільні частини, тобто виконують каскадну реалізацію шляхом порівняння можливих варіантів і вибору з-поміж них оптимального або, принаймні, прийнятнішого. Такі перетворення виконують аналогічно, як і при каскадній реалізації термів (п. 2.3.2), з тією тільки різницею, що спільні частини відокремлюють у кількох функцій. Але шлях подібних перетворень з перебором варіантів може виявитися доволі трудомістким для більш-менш складних схем.

Полегшити розв'язання задачі спільної мінімізації можна застосуванням формалізованих способів, два з яких наводяться нижче.

1. Теоретико-множинний спосіб. Спільні частини функцій можна виокремити шляхом використання співвідношень, виходячи з теоретико-множинних уявлень (п. 6.1.5). Продемонструємо спосіб на прикладі спільної мінімізації сукупності функцій з різними типами зв'язку між ними (таблиця відповідності на рис. 6.14, *а* без колонки *ж*). На рис. 6.14, *б* компактно зображено 12 діаграм термів: 7 для заданих функцій $y_1 \dots y_7$ та 5 для співвідношень між цими функціями та їх частинами.

а) Неперетинні функції. Мінімізуємо функції y_1, y_2 , які є неперетинні, бо, як добре видно з їх діаграм, $K_{12}=y_1y_2=0$ (далі операції між функціями позначатимемо подвійним індексом відповідно до їх нумерації). У цьому випадку згідно з (6.9) для композиції є всього 3 складники: 2 функції, які є невідомі, та диз'юнкція, отже, потрібно мінімізувати два складники і через них

виразити третій. Мінімізуємо першу функцію як найпростішу

$$y_1 = x_1 x_2$$

і диз'юнкцію $D_{12}=y_1+y_2$ (див. діаграму D_{12})

$$\overline{D_{12}} = \overline{x_1 x_2} = \overline{x_1} + \overline{x_2},$$

на основі яких за (6.9) дістаємо вираз другої функції

$$y_2 = \overline{y_1 D_{12}} = \overline{y_1} + \overline{D_{12}}$$

та реалізуємо обидві функції y_1, y_2 спільно на трьох елементах (схемну реалізацію без додаткових посилань див. на рис. 6.14, в).

Якщо вважати функцію y_3 відомою, так само можна мінімізувати неперетинну з нею функцію y_5 . Для цього об'єднанням одиниць з діаграм y_3 і y_5 будуємо діаграму $D_{35}=y_3+y_5$ (див. рис. 6.14, б) та мінімізуємо диз'юнкцію. Неважко помітити, що функція y_2 перетворюється в D_{35} додаванням до неї нижнього рядка одиниць, тобто $D_{35}=y_2+x_3$. Тоді

$$y_5 = \overline{y_3 D_{35}} = \overline{y_3} + \overline{D_{35}},$$

отже, функція y_5 реалізується на двох елементах АБО-НІ.

б) *Включення шуканої функції до складу відомої.* За спільної мінімізації функцій y_2 і y_3 має місце випадок $y_3 \subset y_2$, бо діаграма y_3 є частиною діаграми y_2 . Як і в попередньому випадку, маємо згідно з (6.10) теж 3 складники: y_2, y_3 та $R_{23}=y_2 \setminus y_3$, отже, діємо в такий самий спосіб. Простішу функцію y_2 вже визначено, тому будуємо діаграму для різниці вилученням одиниць із тих клітинок діаграми y_2 , що збігаються з одиницями діаграми y_3 (див. діаграму R_{23}), що еквівалентно вилученню з діаграми y_2 нижнього рядка, тобто x_3 :

$$R_{23} = y_2 \setminus x_3 = \overline{y_2} x_3 = \overline{y_2} + x_3.$$

Відтак, згідно з (6.10) функцію

$$y_3 = y_2 \overline{R_{23}} = y_2 (\overline{y_2} + x_3) = y_2 x_3$$

реалізуємо елементом І.

в) *Включення відомої функції до складу шуканої.* За спільної мінімізації функцій y_3 і y_4 ситуація зворотна поперед-

ній: тепер функція y_3 , яку вважаємо відомою, є частиною шуканої, тобто $y_3 \subset y_4$, що видно безпосередньо з діаграм.

Тому, згідно з (6.11), будуємо діаграму для різниці $R_{43}=y_4 \setminus y_3$, яка збігається з діаграмою y_1 , тобто $R_{43}=y_1$, отже, маємо реалізацію на елементі АБО:

$$y_4 = R_{43} + y_3 = y_1 + y_3.$$

Очевидно, такий самий результат можна дістати, якщо за відому функцію взяти y_1 , бо тоді різниця становитиме $R_{41}=y_4 \setminus y_1=y_3$; отже, $y_4=y_1+R_{41}=y_1+y_3$.

г) *Інверсія*. Безпосередньо з таблиці відповідності видно, що останні дві функції є інверсії вже відомих $y_6 = \overline{y_4}$, $y_7 = \overline{y_5}$ і найпростіше реалізуються за допомогою інверторів. Проте такий спосіб пов'язаний з погіршенням швидкодії (п. 6.4.3). У разі неприйнятності цього утворюють *обернені функції* $y_6^{\circ} = \overline{\overline{y_4}}$, $y_7^{\circ} = \overline{\overline{y_5}}$, які є логічно еквівалентні інверсіям $\overline{y_4}$, $\overline{y_5}$, але формуються окремо, аналогічно прямим функціям. У загальному випадку, коли функції є частково визначеними, вони можуть довизначатися по-різному при синтезі прямих і обернених функцій з метою спрощення виразів. Внаслідок цього вирази для y_6° і y_7° можуть не зводитись один до одного шляхом тотожних перетворень. У цьому й полягає різниця між оберненою й інверсною функціями. Розглянемо синтез функцій y_6 , y_7 як обернених до y_4 , y_5 .

д) *Використання кількох функцій*. Інколи кращий результат спільної реалізації можна дістати, якщо під час синтезу шуканої функції в якості відомої буде деяка композиція (кон'юнкція, різниця тощо) кількох вже відомих функцій. Прикладом є використання такої композиції для синтезу функції y_6 , коли в якості відомої функції є об'єднання $D_{12}=y_1+y_2$ (див. діаграму), яке позначимо $y_0=D_{12}$.

е) *Суцільне покриття області визначення*. При цьому маємо випадок, коли об'єднання шуканої і відомої функцій $D_{06}=y_6+D_{12}=1$ покриває всю діаграму одиницями (див. рис.

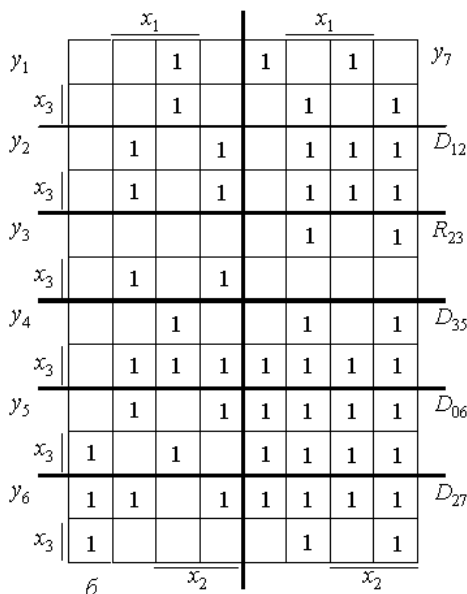
6.14, б), а їх перетин $y_6 D_{12} \neq 0$ відмінний від нуля. На підставі (6.8) визначаємо різницю $R_{06} = D_{12} \setminus y_6$, діаграма якої збігається з діаграмою y_4 , що в свою чергу утворюється об'єднанням y_1 та y_3 . Отже, функція

$$y_6 = {}^{\circ}y_4 = \overline{R_{06}} = \overline{y_1 + y_3}$$

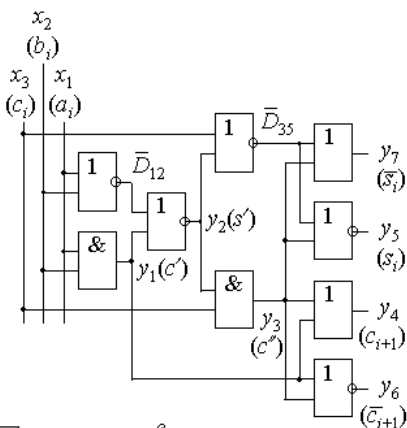
реалізується елементом АБО-НІ.

i	$\begin{matrix} 8 & 4 & 2 \\ x_3 & x_2 & x_1 \end{matrix}$			1							j
	x_3	x_2	x_1	y_1	y_2	y_3	y_4	y_5	y_6	y_7	
0	0	0	0	0	0	0	0	0	1	1	0
1	0	0	1	0	1	0	0	1	1	0	2
2	0	1	0	0	1	0	0	1	1	0	4
3	0	1	1	1	0	0	1	0	0	1	7
4	1	0	0	0	0	0	0	1	1	0	8
5	1	0	1	0	1	1	1	0	0	1	11
6	1	1	0	0	1	1	1	0	0	1	13
7	1	1	1	1	0	0	1	1	0	0	15

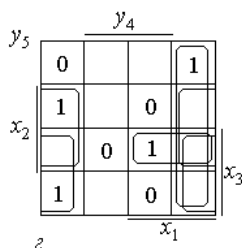
a



б



в



г

Рис. 2.14

Рис. 6.14. Спрощення логічної схеми теоретико-множинним способом

є) *Загальний випадок співвідношення функцій.* Якщо для синтезу функції y_7 скористатися вже реалізованою функцією y_2 , то за ознаками $D_{27}=y_2+y_7 \neq 1$, $K_{27}=y_2y_7 \neq 0$ маємо загальний випадок їх співвідношення. Вибираючи варіант виразу (6.7), синтезуємо D_{27} і K_{27} . Діаграму D_{27} (див. рис. 6.14, б) легко пов'язати з діаграмою y_2 додаванням до останньої верхнього рядка одиниць: $D_{27}=y_2+\overline{x_3}$, а діаграми K_{27} і y_3 збігаються: $K_{27}=y_3$. Отже, з урахуванням $D_{35}=y_2+x_3$ маємо

$$\begin{aligned} y_7 &= {}^{\circ}y_5 = \overline{y_2}D_{27} + K_{27} = \overline{y_2}(y_2 + \overline{x_3}) + y_3 = \\ &= y_2 + x_3 + y_3 = \overline{D_{35}} + y_3, \end{aligned}$$

тобто функція реалізується елементом АБО.

Як і слід було очікувати, обернені функції утворюються так само, як і прямі, якщо останні елементи в них мають взаємно-інверсні виходи. Проте у випадку частково визначених функцій реалізація може бути простішою, якщо останній елемент прямої або оберненої функції має об'єднувати менше термів попередніх ступенів схеми.

Зауважимо, що синтезована схема є суматором (див. табл. 5.10), в дужках на ній наведені прийняті позначення сигналів. Складність синтезованої схеми по прямим виходах y_4 , y_5 або по інверсних виходах y_6 , y_7 становить $q=7/14$.

2. Спосіб із розширенням вхідного кортежу. Другий спосіб полягає в тому, що відома функція вводиться до вхідного кортежу шуканої, тобто розглядається як додаткова її змінна. Після мінімізації отримуємо вираз шуканої функції, аргументом якої крім змінних є ще відома функція, отже, досягається їх каскадна реалізація.

Розглянемо цей спосіб на прикладі спільної мінімізації функцій y_4 , y_5 (рис. 6.14, а). Спочатку за допомогою діаграми (рис. 6.14, б) мінімізуємо функцію y_4 , яку вважаємо простішою

$$y_4 = x_1x_2 + x_1x_3 + x_2x_3 = x_1x_2 + (x_1 + x_2)x_3 = A + Bx_3,$$

де $A=x_1x_2$, $B=x_1+x_2$. Її схема, що є мажоритарним елементом (див. рис. 6.3), має складність $q=4/8$. Розширюючи вхідний кортеж функції $y_5=f(y_4, x_1, x_2, x_3)$, в таблиці відповідності над цими змінними проставляємо вагові коефіцієнти коду $j=x_3x_2x_1y_4$, заповнюємо колонку j , що відображає нумерацію клітинок діаграми (рис. 6.14, з), та вносимо до неї з таблиці значення y_5 . Ця функція виявляється тепер частково визначеною з факультативними значеннями X у порожніх клітинках на неіснуючих у таблиці кортежах j . Дійсно, наприклад, кортеж $j=1$ є неможливий, тому що в суматорі за нульових доданків $x_3=x_2=x_1=0$ не може бути перенесення до старшого розряду $y_4=1$, тому коду $j=1$ відповідає $y_5=X$. Мінімізація дає функцію

$$y_5 = y_4x_1 + y_4x_2 + y_4x_3 + x_1x_2x_3 = \overline{y_4}(B + x_3) + Ax_3,$$

яка з використанням частин A , B функції y_4 має складність $q=5/9$. Отже, спільна реалізація характеризується складністю схеми $q=9/17$ проти $q=7/14$ за попереднім способом.

Обидва розглянуті способи спільної реалізації сукупності функцій можуть доповнювати один одного. При цьому спосіб, що ґрунтується на теоретико-множинних уявленнях, є гнучкішим щодо компонування шуканої функції з уже відомих функцій або їх частин. Способом розширення вхідного кортежу ускладнюється мінімізація за великої кількості змінних. Особливої мінімізації сукупності частково визначених функцій розглядаються в п. 6.4.2.

6.4. Проектування логічних схем

6.4.1. Завдання та етапи логічного проектування

Проектування ЦП полягає у формулюванні технічного завдання, розробці структурної схеми і алгоритмів функціонування пристрою (у разі потреби, також окремих його частин), логічного проектування і, насамкінець, технічного проектування, пов'язаного з розробкою робочих схем та креслень, конструкції, технології, виконання інших заходів щодо підготовки до виготовлення виробу.

Завдання *логічного проектування* полягає у відшуванні такої логічної функції та схеми, яка забезпечує заданий алгоритм функціонування, тобто виконання потрібної функції пристрою за мінімуму необхідного для його реалізації обладнання – кількості елементів, що відповідають певним технічним вимогам. Вибір елементної бази (наступні глави) передуює логічному проектуванню і може коригуватися після нього з метою задовольнити такі вимоги, як швидкодія, споживана потужність, надійність, серійноспроможність, вартість, габарити тощо.

Незважаючи на особливості різних за характером цифрових пристроїв, можна відокремити загальні *етапи*, притаманні формалізованому логічному проектуванню:

- 1) математичне моделювання функціонування пристрою;
- 2) синтез логічної функції, здатної забезпечити алгоритм функціонування, та її мінімізація для скорочення апаратних витрат;
- 3) схемна реалізація, що полягає в перетворенні логічної функції з метою отримати схему пристрою з урахуванням елементної бази.

Охарактеризуємо коротко ці етапи.

На *першому етапі* як підсумок структурного та алгоритмічного проектування пристрою складається його математична модель у вигляді таблиці відповідності, таблиці або діаграми термів, графу переходів, логічної функції тощо. Вихідною математичною моделлю для певного кола типів пристроїв можна вважати ДДНФ або ДКНФ логічної функції, до яких легко перейти від будь-якої форми задавання функціонування пристрою.

Другий етап проектування полягає в синтезі шуканої логічної функції в коротшій формі методом перетворень за допомогою діаграм термів (методом Вайча-Карно) або формалізованими алгебричними методами.

І, нарешті, на *третьому етапі* вирішується завдання перетворень МДНФ (МКНФ) із застосуванням способів схемної мінімізації до функції в будь-якій формі, зокрема, мішаній, зручній для побудови пристрою з урахуванням вибраної елементної бази. Здобуту таким чином формулу часто називають *структур-*

ною логічною функцією, а цей етап – її схемною реалізацією.

Таким чином, логічне проектування полягає в здобутті структурної функції, яка відображає логічну схему, здатну забезпечити заданий алгоритм функціонування пристрою при мінімумі обладнання.

6.4.2. Методика проектування логічних схем

Залежно від компонентів, на яких реалізується логічна схема, розрізняють логічне моделювання на різних рівнях. Тут ми розглядаємо проектування *на рівні логічних елементів* (вентильному рівні), тобто на ІС малого ступеня інтеграції. Крім того, обмежуємося поки що лише схемами без зворотних зв'язків, хоч основні принципи проектування застосовуються для будь-яких схем. Вважатимемо також, що логічному проектуванню передують як визначення в будь-якій формі функціонування проектованого пристрою, так і вибір елементної бази з урахуванням технічних характеристик пристрою та його компонентів. Методику логічного проектування розглянемо окремо для схем з одним виходом та з кількома виходами [12].

1. Схеми з одним виходом. Припустимо, що потрібно реалізувати частково визначену функцію y_1 (рис. 6.15, а) в елементному базисі І-НІ. Проектування доцільно виконувати в такому порядку.

а) За таблицею відповідності для функції y_1 будуємо діаграму термів (на рис. 6.15, б компактно подано шість діаграм), до якої вносимо тільки нулі та одиниці, тоді порожнім клітинкам відповідатимуть факультативні значення X .

б) Мінімізуємо логічну функцію, застосовуючи розглянуті прийоми спрощення (п. 6.3.2). У прикладі, по-перше, для утворення найкращих об'єднань довизначаємо клітинку з кодом $i=12$ нулем, а всі інші порожні клітинки – одиницями (колонка довизначеної функції відображена в таблиці в дужках). По-друге, застосовуємо редукцію, вважаючи спочатку клітинки $i=14, 15$ одиничними, а відтак, коригуємо функцію вилученням цих клітинок

$$y_1 = (x_1 + x_2) \setminus x_2 x_3 x_4 = (x_1 + x_2) \cdot \overline{x_2 x_3 x_4}. \quad (6.26)$$

Інший шлях спрощення полягає у виборі 3-поміж мінімальних форм такої, яка має меншу складність реалізації та в якій інверсії змінних, за можливістю, зосереджені в окремих термах, а не розподілені між ними. У нашому випадку такою є МКНФ, яку отримуємо за сполуками нулів

$$y_1 = (x_1 + x_2)(\overline{x_2} + \overline{x_3} + \overline{x_4}) \quad (6.27)$$

і яка зводиться до (6.26).

в) Шляхом тотожних перетворень здійснюємо подальше спрощення та перехід до структурної функції з урахуванням елементної бази. У прикладі розкриттям дужок переходимо від КФ (6.26) до ДФ, а відтак – до базису І-НІ:

$$y_1 = x_1 A + x_2 A = \overline{BC}, \quad (6.28)$$

де $A = \overline{x_2 x_3 x_4}$; $B = \overline{x_1 A}$; $C = \overline{x_2 A}$.

г) За отриманою структурною функцією будуємо схему (на рис. 6.15, в частина схеми по виходу y_1) та оцінюємо ефективність і, отже, доцільність виконаного її спрощення порівнянням складності, що за (6.28) становить $q=4/9$ проти $6/10$ за первісним по мінімізації виразом без застосування редукції.

д) Визначаємо основні параметри (п. 6.4.3): швидкодію, яка в прикладі через триступеневу реалізацію оцінюється затримкою $t_y=3t_{3,п}$; максимальний коефіцієнт об'єднання, що становить для одного елемента $m=3$, а для трьох інших $m=2$; навантаження на джерела сигналів, яке не перевищує розгалуження на $k=2$ входи, а також інші технічні параметри (споживана потужність, напруга живлення тощо), що визначаються характеристиками елементної бази.

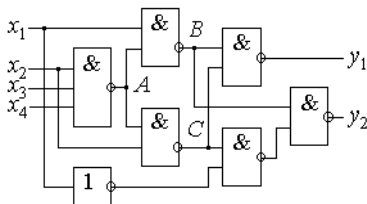
2. Схеми з кількома виходами. Порядок логічного проектування розглянемо на прикладі схеми з двома виходами y_1 , y_2 , що задана таблицею відповідності (див. рис. 6.15, а).

а) Попередньо мінімізуємо логічні функції окремо, порівнюємо їх у булевому базисі та вибираємо одну з простіших, бажано таку, яка за деякими фрагментами логічного виразу або за розташуванням нулів чи одиниць на діаграмі наближається до інших. У прикладі функцію y_1 визначено виразом (6.28), а функ-

$$\overline{y_2} = \overline{x_2} + \overline{x_1 x_3} + \overline{x_1 x_4} + \overline{x_1 x_3 x_4} = \overline{x_2 x_1 x_1 x_3 x_4} \cdot \overline{x_3 x_4 x_1 x_3 x_4} \quad (6.29)$$

i	x_4	x_3	x_2	x_1	y_1	y_2	K	R_{21}
0	0	0	0	0	0	\mathbb{X}	0	\mathbb{X}
1	0	0	0	1	$\mathbb{X}(1)$	$\mathbb{X}(1)$	$\mathbb{X}(1)$	0
2	0	0	1	0	$\mathbb{X}(1)$	$\mathbb{X}(0)$	$\mathbb{X}(0)$	0
3	0	0	1	1	$\mathbb{X}(1)$	$\mathbb{X}(1)$	$\mathbb{X}(1)$	0
4	0	1	0	0	0	1	0	1
5	0	1	0	1	$\mathbb{X}(1)$	$\mathbb{X}(1)$	$\mathbb{X}(1)$	0
6	0	1	1	0	1	0	0	0
7	0	1	1	1	$\mathbb{X}(1)$	1	1	0
8	1	0	0	0	0	1	0	1
9	1	0	0	1	$\mathbb{X}(1)$	$\mathbb{X}(1)$	$\mathbb{X}(1)$	0
10	1	0	1	0	$\mathbb{X}(1)$	0	0	0
11	1	0	1	1	1	1	1	0
12	1	1	0	0	$\mathbb{X}(0)$	\mathbb{X}	0	\mathbb{X}
13	1	1	0	1	1	1	1	0
14	1	1	1	0	0	1	0	1
15	1	1	1	1	0	0	0	0

6



6

y_1	y_2	D	K	R_{12}	R_{21}	M
0	0	0	0	0	0	0
0	1	1	0	0	1	1
1	0	1	0	1	0	1
1	1	1	1	0	0	0

2

Рис. 6.15. Проектування схеми з кількома виходами

б) З'я'овуємо, чи не є співвідношення між шуканою фун-

кцією y_2 та функцією y_1 , яку вважаємо відомою, окремим випадком (п. 6.1.5). Безпосередньо з діаграм видно, що їх об'єднання $D \neq 1$, перетин $K \neq 0$, тому маємо загальний випадок (6.7). При реалізації схеми, наприклад, в базисі І-НІ всі логічно еквівалентні варіанти виразу (6.7) приводять до однакового результату, тому з-поміж них слід вибрати найпростіший для мінімізації і перетворень. З порівняння діаграм у прикладі вибираємо пару K і R_{21} (див. рис. 6.15, б).

в) За діаграмами мінімізуємо вирази результатів вибраних операцій.

Особливістю спільної мінімізації частково визначених функцій є необхідність забезпечення *сумісності операцій* як між собою, так і з функціями, над якими вони виконуються, тому що під час довизначення порожніх клітинок на діаграмах можуть утворитися неможливі співвідношення, що суперечать таблиці відповідності (рис. 6.15, з).

Аби уникнути цього, доцільно діяти в такому порядку. Вважаємо спочатку функцію y_1 відомою і повністю визначеною (нагадаємо, що тепер порожні клітинки на її діаграмі, крім $i=12$, відповідають лог. 1), а функцію y_2 – частково визначеною (основний її стовпець у таблиці). До стовпця перетину $K=y_1y_2$ переносимо всі нулі із зазначених колонок y_1 і y_2 та лише ті одиниці, що збігаються в порівнюваних стовпцях, а комбінації $K=y_1y_2=1X=X$ залишаються невизначеними. Далі заповнюємо діаграму K та довизначаємо порожні клітинки для утворення найкращих об'єднань.

Метою мінімізації є утворення найпростішої функції з використанням відомих функцій або їх частин. Якщо на діаграмі K клітинку $i=2$ довизначити нулем, а інші одиницями (подано в дужках біля стовпця K таблиці), то повністю повторюється діаграма \overline{B} , тобто

$$K = \overline{B}. \quad (6.30)$$

Цього висновку можна дійти й аналітично. Три сполуки з чотирьох одиниць кожна, зважаючи на (6.28), дають вираз

$$K = x_1\overline{x_2} + x_1\overline{x_3} + x_1\overline{x_4} = x_1(\overline{x_2} + \overline{x_3} + \overline{x_4}) = x_1\overline{x_2x_3x_4} = x_1A = \overline{B}.$$

Таким довизначенням кон'юнкції автоматично довизначається й частина функції y_2 (наведено в дужках біля її стовпця таблиці): $y_2=K$, якщо $y_1=1$, а на наборах $i=0$ та 12 , що відповідають $y_1=K=0$, вона лишається невизначеною.

У такий самий спосіб мінімізуємо й результат другої операції $R_{21}=y_2 \setminus y_1$. До стовпця R_{21} переносимо з колонки y_2 тільки ті одиниці, що відповідають $y_1=0$, а інші значення виявляються нульовими, крім тих, де $y_2=X$, $y_1=0$, які залишаються невизначеними (для $i=0$ та 12). Якщо довизначити їх одиницями на діаграмі R_{21} , отримаємо діаграму C , з якої вирізано смугу одиниць завширшки x_1 , тобто

$$R_{21} = C \setminus x_1 = \overline{C x_1}. \quad (6.31)$$

Аналітично це можна виразити з двох сполук одиниць з урахуванням (6.28):

$$\begin{aligned} R_{21} &= \overline{x_1 x_2} + \overline{x_1 x_3 x_4} = \overline{x_1 (x_2 + x_3 x_4)} = \overline{\overline{x_1 x_2 x_3 x_4}} = \\ &= \overline{\overline{x_1 x_2 x_3 x_4}} = \overline{x_1 x_2} A = \overline{x_1} C. \end{aligned}$$

Тепер решту значень X у стовпцях R_{21} та y_2 довизначено до $X=1$.

Таким чином, довизначаємо й мінімізуємо не саму функцію y_2 , а співвідношення між функціями, у нашому випадку K та R_{21} . Якщо для спрощення виразів цих операцій бажано інакше довизначити функцію y_1 без істотного її ускладнення, розглядаємо варіанти і вибираємо найкращий.

2) Завершуємо проектування аналогічно схемі з одним виходом. У прикладі згідно з (6.7), (6.30), (6.31) дістаємо структурну функцію

$$y_2 = K + R_{21} = \overline{B} + \overline{x_1} C = \overline{\overline{B x_1 C}}$$

та реалізуємо її в базисі І-НІ (рис. 6.15, в). Ефективність виконаного проектування спільною мінімізацією функцій оцінюємо складністю схеми $q=7/14$ проти $8/20$ при окремій їх реалізації.

У варіанті з використанням операції додавання за модулем два з (6.7) маємо $y_2=y_1 \oplus M$. Неважко переконатися, що при довизначенні функції як у прикладі, отримаємо $M=\overline{y_1} \oplus y_2=x_1$.

Отже, функція

$$y_2 = y_1 \oplus M = y_1 \oplus \overline{x_1} = \overline{y_1 \oplus x_1}$$

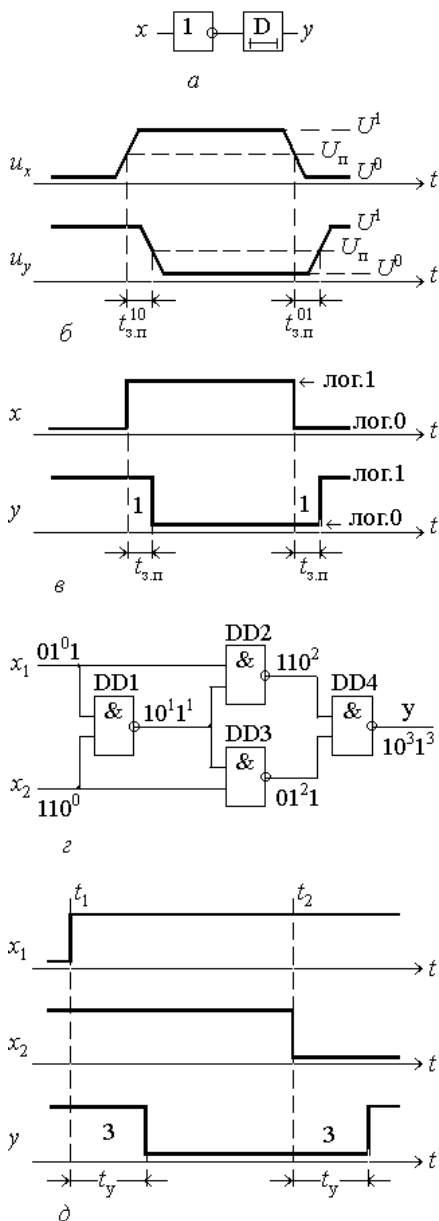
реалізується лише одним додатковим елементом І-АБО-НІ.

6.4.3. Елементи технічного проектування

Вже на стадії логічного проектування потрібно враховувати можливості елементної бази щодо схемної реалізації синтезованої структурної функції. Це врахування може спричинити перетворення функції до такого виразу, який технічно можна втілити на ІС вибраної серії. Крім виконуваних елементами серії логічних функцій доводиться зважати й на такі особливості, які стосуються технічного проектування. Розглянемо основні з них.

1.ахування швидкодії логічних елементів. Логічний елемент умовно можна моделювати двома частинами: безінерційною, що виконує логічну функцію, наприклад, інверсію (рис. 6.16, а), та елементом затримки D (Delay – затримка), який відображає перехідні процеси (рис. 6.16, б) усталення вихідного сигналу. Тривалість перехідних процесів відлічується відносно порогової напруги $U_{\text{п}}$ спрацювання елемента й оцінюється часом затримки поширення сигналу $t_{\text{з.п}}^{10}$ під час переходу вихідної напруги від рівня лог. 1 – U^1 до рівня лог. 0 – U^0 , а також часом затримки поширення $t_{\text{з.п}}^{01}$ під час перемикання в протилежному напрямку. За ланцюжкового сполучення елементів зручно користуватися усередненим параметром – середнім часом затримки поширення $t_{\text{з.п}} = 0,5(t_{\text{з.п}}^{10} + t_{\text{з.п}}^{01})$, а часові діаграми зображати ідеалізовано (рис. 6.16, в). Часові інтервали на них позначатимемо цифрами, які вказують на тривалість у кількості середніх затримок $t_{\text{з.п}}$.

Умовна дискретизація перехідних процесів у часі дозволяє спростити підрахунок швидкодії ЦП до виразу: $t_y = N t_{\text{з.п}}$, де t_y – час усталення сигналу на виході пристрою, N – максимальна кількість однотипних елементів на шляху поширення його від входу до виходу. Наприклад, час усталення в елементі виняткове АБО (див. рис. 6.13, з) визначається затримкою трьох елементів І-НІ. Для наочності шлях поширення сигналу позначатимемо його двійковими кодами безпосередньо на схемі (рис.



6.16, з). У першій позиції кодів наведено початковий стан схеми при $x_1=0$, $x_2=1$. У другій позиції нульовим індексом вгорі $x_1=1^0$ зазначено, що від зміни цього рівня на вході починається відлік затримки поширення сигналу на шляху: $x_1 \rightarrow DD1 \rightarrow DD3 \rightarrow DD4 \rightarrow y$, тому на виході елементу DD1 індексом 1 позначено одну затримку $t_{з.п}$, на виході DD3 – дві затримки і, нарешті, на виході y – три затримки. На часових діаграмах (рис. 6.16, д) цей процес відображено, починаючи з моменту t_1 .

Аналогічно відбувається перехідний процес при зміні рівня на другому вході до $x_2=0$, як зазначено кодами в третій позиції станів і на діаграмах в момент t_2 . Отже, час усталення пристрою під час перемикання в обох напрямках становить $t_y=3t_{з.п}$.

При зображенні логічного виразу в універсальних базисах І-НІ чи АБО-НІ час усталення можна підрахувати за найбільшою кількістю інверсій над аргументами. Так, безпосе-

Рис. 6.16. До визначення швидкодії логічного елементу 229

редньо з (6.24) за трьома інверсіями встановлюємо, що час поширення сигналу визначається затримкою трьох елементів І-НІ.

Таким чином, час усталення ЦП визначається не тільки швидкодією елементів, але й *глибиною реалізації схеми*. Реалізація за первісними після мінімізації термами є двоступеневою (рис. 6.13, *ж*), якщо вхідні сигнали вважати двофазними, отже, має найвищу швидкодію. Спрощення схеми шляхом каскадування поряд із перевагою (ощадливість обладнання) призводить одночасно до погіршення швидкодії. Так, схема на рис. 6.13, *з* має більшу глибину реалізації і є тріступеневою. Якщо каскадна реалізація не задовольняє вимоги швидкодії, доводиться повертатися до двоступеневої реалізації з більшою складністю схеми.

2. Врахування коефіцієнта об'єднання. Якщо в елементі бракує входів для реалізації терма, вдаються до багатоступневих (деревоподібних) схем нарощування коефіцієнта об'єднання. Елементи без інверсії на виході каскадуються за схемою на рис. 6.17, *а*, де позначка * вказує на двомісну операцію І, АБО чи суми за модулем два. Так, коефіцієнт об'єднання $m=7$ на двовходових елементах І утворюється за формулою:

$$y = DE = (AB)(Cx_7) = (x_1x_2)(x_3x_4)(x_5x_6)x_7,$$

яка потребує $m-1$ елементів.

Принцип об'єднання елементів з інверсією на виході (І-НІ, АБО-НІ) для виконання логічної функції, однойменної з назвою елементу (рис. 6.17, *б*), покажемо на прикладі операції І-НІ для $m=5$:

$$y = \overline{ABx_5}, \quad A = \overline{x_1x_2}, \quad B = \overline{x_3x_4},$$

де подвійна риска означає, що одна інверсія виконується елементом І-НІ, а друга – додатковим інвертором згідно з рис. 6.10, *б*, 6.11, *б*. При виконанні функції, протилежної назві елемента (у базисі І-НІ виконується функція АБО-НІ чи в базисі АБО-НІ – функція І-НІ), об'єднання ще більш ускладнюється (рис. 6.17, *в*), наприклад,

$$y = \overline{x_1x_2x_3x_4x_5} = \overline{A+B+x_5}, \quad A = \overline{x_1+x_2}, \quad B = \overline{x_3+x_4}.$$

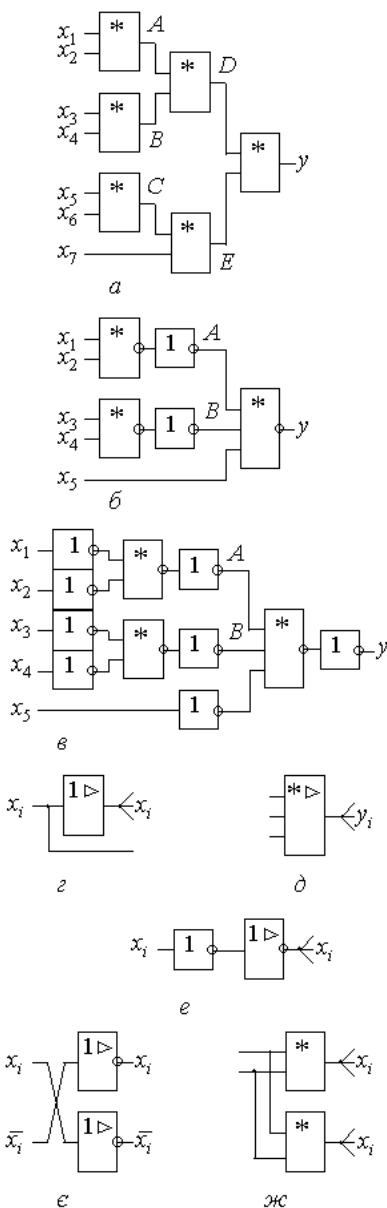
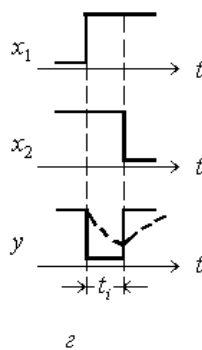
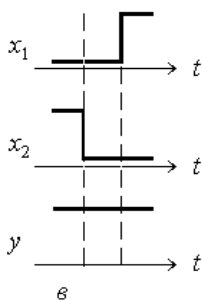
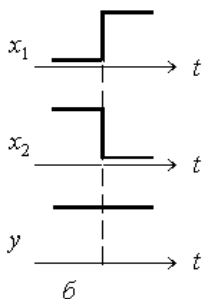
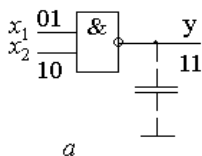


Рис. 6.17. Каскадування логічних елементів

Багатоступеневими схемами можна реалізувати довільні коефіцієнти об'єднання, проте з погіршенням швидкодії і додатковими витратами елементів. Цей недолік послаблюється застосуванням модифікованих елементів з розширеними функційними можливостями або реалізацією у мішаному базисі шляхом перетворення функцій.

3. Врахування коефіцієнта розгалуження. Навантаження на кожне джерело зовнішнього входного сигналу або на елемент-джерело внутрішньої змінної, яка є функцією зовнішніх аргументів, легко підраховується за кількістю входжень тієї чи іншої змінної до структурної функції проектованої логічної схеми. Якщо ця кількість перевищує допустимий коефіцієнт розгалуження джерела, проводять заходів щодо його розвантаження.

Передусім, за допомогою додаткового елемента на вході, який відіграє роль підсилювача (рис. 6.17, г), розв'яують джерело сигналу від навантажувальних елементів, а всередині проектованої схеми при цьому замість звичайного застосовують елемент з підви-



щеною навантажувальною здатністю (рис. 6.17, д). Якщо використовують підсилювальні елементи з інверсним виходом, відновлення аргументу здійснюють шляхом подвійного інвертування (рис. 6.17, е), а у випадку парафазних сигналів достатньо й одного інвертування (рис. 6.17, є). Для запобігання погіршенню швидкодії через затримку в додаткових розвантажувальних елементах використовують спосіб дублювання елемента (рис. 6.17, ж), якщо він сам не перевантажує сигнал.

4. ахування небезпечних змагань сигналів. З огляду на неоднакові затримки в попередніх елементах та різні шляхи поширення виникають так звані змагання сигналів: вони потрапляють до входів пристрою неодноразово, внаслідок чого в ньому можуть утворюватися сигнали, не передбачені при правильній роботі пристрою – *ризик*, зокрема, імпульсні завади. У такому разі змагання є небезпечними. Якщо, наприклад, до входів елемента І-НІ (рис. 6.18, а) перепади сигналів x_1 , x_2 потрапляють одночасно (рис. 6.18, б) або негативний фронт випереджає позитивний (рис. 6.18, в), то на виході у ризиків відсутні, а якщо, навпаки, позитивний фронт випереджає негативний, виникає ризик – імпульсна завада тривалістю t_i (рис. 6.18, г), яка може спричинити хибне спрацювання навантажувальних елементів.

Для усунення небезпечних змагань можна передбачити такий порядок зміни рівнів, коли паразитний викид стає дуже коротким або не виникає зовсім, наприклад, штучною затримкою

Рис. 6.18. Врахування небезпечних змагань сигналів

сигналу x_1 , як на рис. 6.18, *в*. Проте у відносно складних ЦП запобігти ризикам таким чином важко, тому вдаються до схемних засобів їх усунення. Якщо в окремі моменти часу можлива зміна лише одного вхідного сигналу, то є змога виконати логічну схему, вільну від небезпечних змагань. Умовою цього є зв'язність сполук на діаграмі термів: усі сусідні одиничні клітинки мають бути об'єднаними хоча б однією сполукою. Ця умова виконується, наприклад, на діаграмі рис. 6.13, *є*, а на рис. 6.5, *в* слід ввести дві додаткові сполуки, зображені пунктиром. Проте найчастіше застосовують синхронізацію пристроїв: спрацьовування елементів визначаються моментами надходження синхроімпульсів, період яких вибирається таким, щоб перехідні процеси усталення рівнів встигли закінчитись. Згідно з рис. 6.18, *г* зчитуванням вхідної інформації синхроімпульсом після інтервалу t_i можна позбутися паразитного викиду.

Заваду можна зменшити до припустимого рівня також за допомогою згладжувального фільтра типу інтегрувального кола (пунктир на рис. 6.18, *а*). Проте через погіршення швидкодії такий спосіб використовується рідко, в основному для виявлення місця виникнення завади: якщо з приєднанням до виходу у невеликої ємності викид зменшується (пунктир на рис. 6.18, *г*), слід усунути змагання сигналів на входах елементу.

6.5. Приклад логічного проектування

1. Мінімізуємо функцію $y = \overline{\overline{x_2(x_1 \oplus x_4) + x_2 x_3 + x_4}}$ за допомогою логічних перетворень, складемо таблицю відповідності та отримаємо співвідношення для ДДНФ і ДКНФ.

$$y = \overline{\overline{x_2(x_1 \oplus x_4) + x_2 x_3 + x_4}} = \overline{A + B};$$

$$A = \overline{\overline{x_2(x_1 \oplus x_4)}} = \overline{\overline{x_2 x_1 x_4} + \overline{x_1 x_2 x_4}};$$

$$B = \overline{\overline{x_2 x_4 + x_3}} = \overline{\overline{x_2 x_3 x_4}};$$

$$\begin{aligned} \overline{A + B} &= \overline{\overline{\overline{x_2 x_1 x_4} + \overline{x_1 x_2 x_4}} + \overline{\overline{x_2 x_3 x_4}}} = \overline{\overline{x_2 x_1 x_4} \cdot \overline{x_1 x_2 x_4} \cdot \overline{\overline{x_2 x_3 x_4}}} = \\ &= (\overline{x_1} + x_2 + x_4)(\overline{x_1} + x_2 + \overline{x_4})(\overline{x_2} + x_3 + x_4). \end{aligned}$$

i	x_4	x_3	x_2	x_1	y
0	0	0	0	0	1
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	0
4	0	1	0	0	1
5	0	1	0	1	0
6	0	1	1	0	1
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	1
10	1	0	1	0	1
11	1	0	1	1	1
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	1

Для отриманої мінімізованої форми методом прямої індукції складемо таблицю відповідності.

За даними таблиці складемо до-
сконалі диз'юнктивну та кон'юнктивну
нормальні форми:

- ДДНФ:

$$y = \bar{x}_4 \bar{x}_3 \bar{x}_2 \bar{x}_1 + \bar{x}_4 x_3 \bar{x}_2 \bar{x}_1 + \bar{x}_4 x_3 x_2 \bar{x}_1 + \\ + \bar{x}_4 x_3 x_2 x_1 + x_4 \bar{x}_3 \bar{x}_2 \bar{x}_1 + x_4 \bar{x}_3 x_2 \bar{x}_1 + \\ + x_4 \bar{x}_3 x_2 x_1 + x_4 x_3 \bar{x}_2 \bar{x}_1 + x_4 x_3 x_2 \bar{x}_1 + \\ + x_4 x_3 x_2 x_1;$$

- ДКНФ:

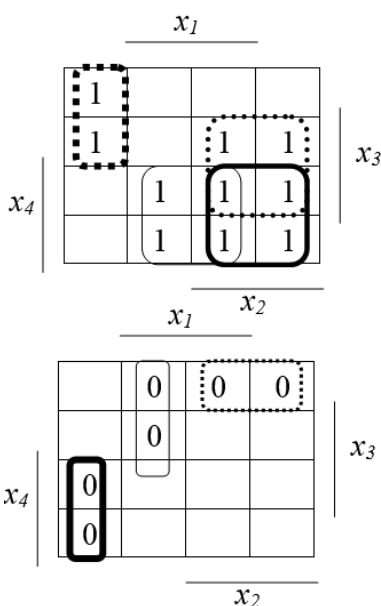
$$y = (x_4 + x_3 + x_2 + \bar{x}_1) + (x_4 + x_3 + \bar{x}_2 + x_1) + \\ + (x_4 + x_3 + \bar{x}_2 + \bar{x}_1) + (x_4 + \bar{x}_3 + x_2 + \bar{x}_1) + \\ + (\bar{x}_4 + x_3 + x_2 + x_1) + (\bar{x}_4 + \bar{x}_3 + x_2 + x_1).$$

2. Мінімізуємо функцію в

диз'юнктивній формі для прямої
функції. Для цього складемо діа-
граму термів (карту Карно), на
якій позначимо у відповідних
клітинках одиниці і, об'єднуючи
їх у об'єднання, складемо
МДНФ:

$$y = x_1 x_4 + x_2 x_4 + x_2 x_3 + \\ + \bar{x}_1 \bar{x}_2 \bar{x}_4. \quad (6.32)$$

Користуючись тією ж ді-
аграмою термів, складемо
МКНФ для інверсної функції,
при цьому використовуються ті
ж об'єднання одиниць, але утво-
рюється кон'юнктивна форма
(дана форма отримується при



взаємній заміні операцій І та АБО, а також інвертуванні всіх

змінних):

$$\bar{y} = (\bar{x}_1 + \bar{x}_4)(\bar{x}_2 + \bar{x}_4)(\bar{x}_2 + \bar{x}_3)(x_1 + x_2 + x_4). \quad (6.33)$$

При мінімізації функції в кон'юнктивній формі для прямої функції в діаграмі термів будемо об'єднувати нулі і, відповідно до цих об'єднань, складемо МКНФ:

$$y = (x_1 + x_2 + \bar{x}_4)(\bar{x}_1 + x_2 + x_4)(\bar{x}_2 + x_3 + x_4). \quad (6.34)$$

На основі цієї ж діаграми термів утворюємо МДНФ для інверсної функції (дана форма аналогічно отримується при взаємній заміні операцій І та АБО, а також інвертуванні всіх змінних):

$$\bar{y} = \bar{x}_1 \bar{x}_2 x_4 + x_1 \bar{x}_2 \bar{x}_4 + x_2 x_3 \bar{x}_4. \quad (6.35)$$

3. Для реалізації функції в булевих базисах (І-НІ, АБО-НІ, І-АБО-НІ) спочатку запишемо функції в даних базисах, а потім порівняємо складність реалізації функції у цих базисах. При цьому необхідно пам'ятати, що в базисі І-НІ реалізуються пряма (6.32) або інверсна (6.35) МДНФ, в базисі АБО-НІ – пряма (6.34) або інверсна (6.33) МКНФ, а в базисі І-АБО-НІ – інверсна (6.35) або пряма (6.32) МДНФ. Для наочності результати зведемо до таблиці 6.4.

Таблиця 6.4. Порівняння складності мінімальних форм

Базис	Вид функції у даному базисі	Складність
І-НІ	$y = x_1 x_4 \bar{x}_2 x_4 \bar{x}_2 x_3 x_1 x_2 x_4$	$q = 8/16$
	$y = x_1 \bar{x}_2 x_4 x_1 x_2 \bar{x}_4 \bar{x}_2 x_3 x_4$	$q = 9/17$
АБО-НІ	$y = x_1 + x_2 + x_4 + x_1 + x_2 + x_4 + x_2 + x_3 + x_4$	$q = 7/15$
	$y = x_1 + x_4 + x_2 + x_4 + x_2 + x_3 + x_1 + x_2 + x_4$	$q = 10/18$
І-АБО-НІ	$y = x_1 \bar{x}_2 x_4 + x_1 x_2 \bar{x}_4 + x_2 x_3 \bar{x}_4$	$q = 5/13$
	$y = x_1 x_4 + x_2 x_4 + x_2 x_3 + x_1 x_2 \bar{x}_4$	$q = 5/13$

З таблиці видно, що найменшу складність має реалізація у базисі І-АБО-НІ, в базисі І-НІ мінімальна складність дорівнює $q = 8/16$, а в базисі АБО-НІ – $q = 7/15$. Реалізації заданої логічної функції в різних елементних базисах показані на рис. 6.19.

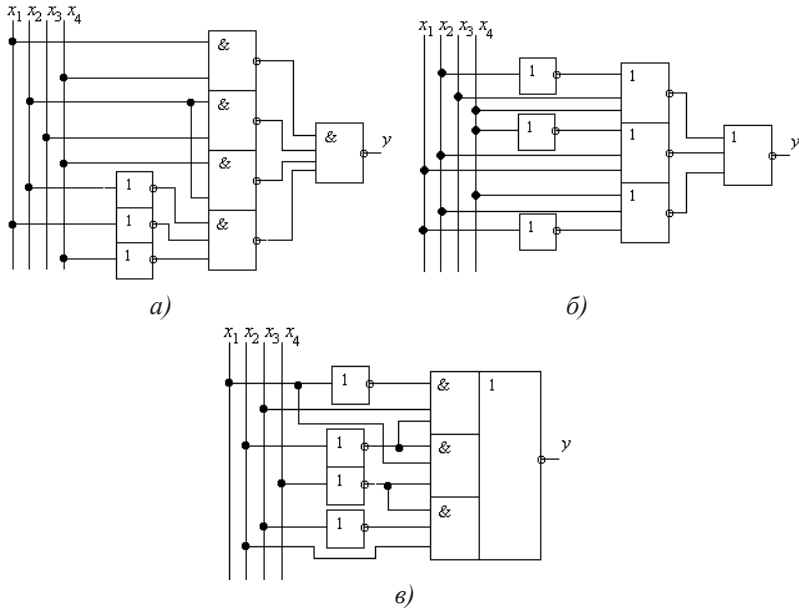
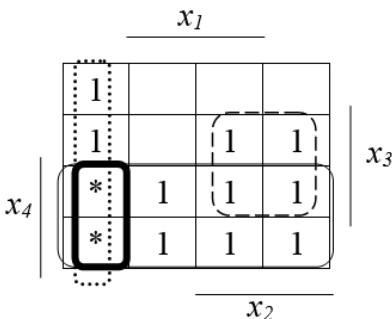


Рис. 6.19. Схемні реалізації функції у булевих базисах:
І-НІ (а); АБО-НІ (б); І-АБО-НІ (в)

4.зглянемо варіанти схемної мінімізації (редукції) в різних елементних базисах.



Спочатку спробуємо мінімізувати схему в базисі І-НІ. Для цього складемо діаграму термів для МДНФ, у якій деякі нулі будемо вважати одиницями (позначені “*”), при цьому функція запишеться так:

$$y = y_1 / y_2 = y_1 \overline{y_2},$$

де y_1 – функція, утворена в результаті об’єднання одиниць і “*”;
 y_2 – допоміжна функція, утворена в результаті об’єднання “*”.

За розглянутою картою Карно визначимо відповідні функції:

$$y_1 = x_4 + \overline{x_2} x_1 + x_3 x_2;$$

$$y_2 = x_1 x_2 x_4.$$

Тоді шукана функція y буде мати вигляд:

$$y = y_1 y_2 = (x_4 + \overline{x_2} x_1 + x_3 x_2) \overline{x_1} x_2 x_4.$$

Відповідно до розподільчого закону функцію можна переписати таким чином:

$$y = x_4 x_1 x_2 x_4 + x_2 x_1 x_1 x_2 x_4 + x_3 x_2 x_1 x_2 x_4.$$

Перетворимо функцію до базису І-НІ:

$$y = x_4 x_1 x_2 x_4 x_2 x_1 x_1 x_2 x_4 x_3 x_2 x_1 x_2 x_4.$$

Складність реалізації для останньої функції дорівнює $q = 7/16$, тобто для базису І-НІ отримали деяке спрощення (див. табл. 6.4).

		x_1		
		$\overline{x_1}$	x_1	x_1
x_4	$\overline{x_3}$	*	0	0
	x_3	*	0	
	$\overline{x_2}$	0		
	x_2	0		
		x_2		

Для мінімізації схеми в базисі АБО-НІ використовують формулу:

$$y = y_1 - y_2 = y_1 + y_2,$$

де y_1 – функція, утворена в результаті об’єднання нульових клітин та клітин, позначених “*”; y_2 – допоміжна функція утворення в результаті об’єднання клітин, позначених “*”.

Визначимо ці функції:

$$y_1 = (x_1 + x_2)(x_2 + x_4)(x_3 + x_4);$$

$$y_2 = (x_1 + x_2 + x_4).$$

Шукана функція y в такому разі дорівнює

$$y = (x_1 + x_2)(x_2 + x_4)(x_3 + x_4) + x_1 + x_2 + x_4.$$

Відповідно до розподільчого закону функцію можна переписати таким чином:

$$y = (x_1 + x_2 + \overline{x_1 + x_2 + x_4})(x_2 + x_4 + \overline{x_1 + x_2 + x_4}) \times \\ \times (x_3 + x_4 + \overline{x_1 + x_2 + x_4}).$$

Перетворимо функцію до базису АБО-НІ і підрахуємо її складність реалізації:

$$y = x_1 + x_2 + x_1 + x_2 + x_4 + x_2 + x_4 + x_1 + x_2 + x_4 + x_3 + x_4 + x_1 + x_2 + x_4, \\ q = \frac{5}{15}.$$

Тому що складність реалізації в базисі АБО-НІ менша, ніж у базисі І-НІ (отриманий результат менший від будь-якого, наведеного в таблиці 6.4 для базисів І-НІ чи АБО-НІ), то схему реалізацію доцільно проводити в базисі АБО-НІ (рис.6.20).

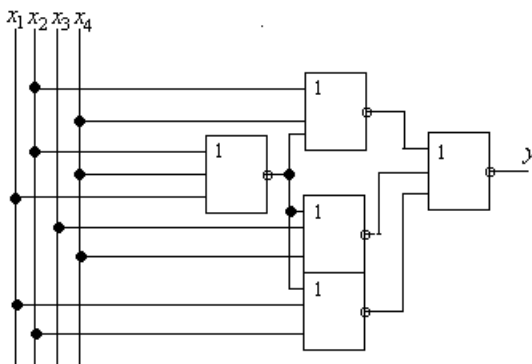


Рис. 6.20. Схемна реалізація мінімізованої логічної функції в базисі АБО-НІ

Для технічного проектування вибираємо мінімізовану схему в базисі АБО-НІ, зображену на рис. 6.20.

Для реалізації даної схеми необхідно підібрати мікросхеми так, щоб можна було реалізувати схему на п'яти тривходових елементах АБО-НІ (ЗАБО-НІ) (рис. 6.21).

Виходячи з вищевказаної умови, вибираємо дві мікросхеми 555ЛЕ4. Усі необхідні для проектування параметри

мікросхеми показані у таблиці 6.5.

Таблиця 6.5. Параметри вибраної мікросхеми

Тип ІМС	Напруга живлення $U_{ж}$, В	Споживаний струм, I_{max} , мА		Затримка поширення $t_{мз,п}$, нс	
		Всі входи в “1”	Всі входи в “0”	Всі входи в “1”	Всі входи в “0”
555ЛЕ4	5	1,8 – 3,6	6,9 – 13,8	10 – 15	10 – 15

Цоколювка вибраної ІМС представлена на рис. 6.21.

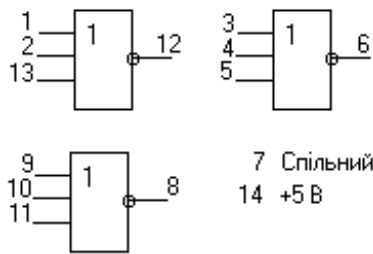


Рис. 6.21. Цоколювка мікросхеми 555ЛЕ4

Результати аналізу (моделювання) мінімізованої схеми (рис. 6.20) на ЕОМ наведені на рис. 6.22, рис. 6.23 та рис. 6.24.

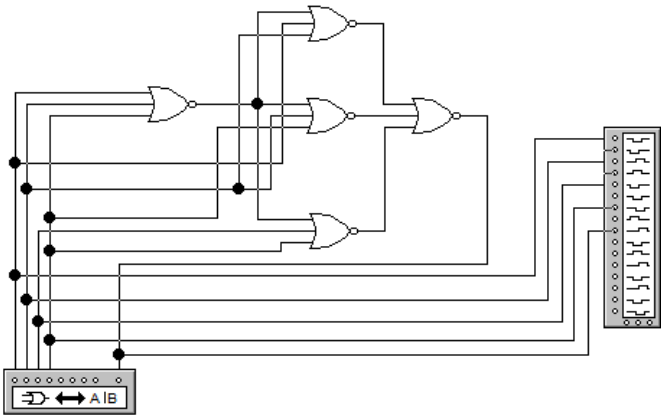


Рис. 6.22. Модель досліджуваної схеми

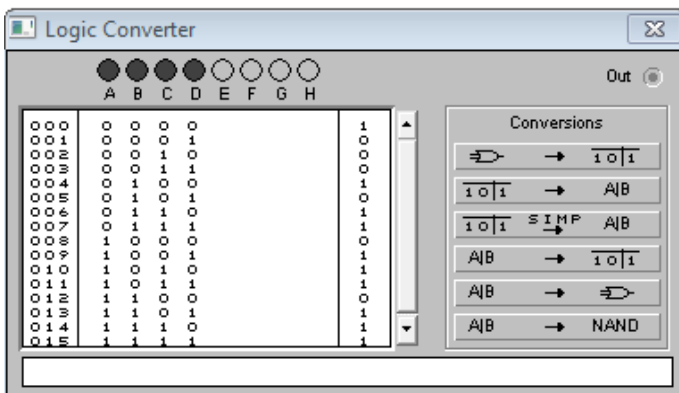


Рис. 6.23. Утворена за допомогою Logic Converter таблиця відповідності

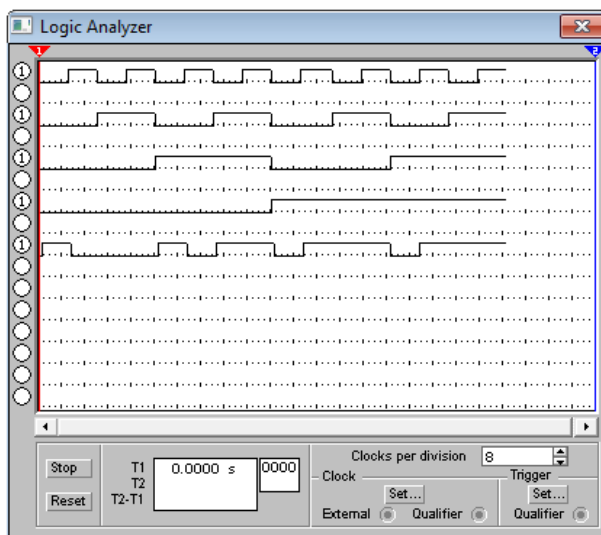


Рис. 6.24. Утворена за допомогою Logic Analyzer діаграми роботи мінімізованої схеми

Таким чином, проведене моделювання підтвердило правильність проведених розрахунків, оскільки результати, отримані в результаті моделювання схеми, повністю відповідають теоретичним розрахункам.

Запитання та вправи

6.1. Які системи логічних функцій є функціонально повні? Які з них є мінімально повні? Доведіть, що мінімально повну систему утворюють: 1) операція заборони і константа одиниці; 2) операція імплікації і константа нуля.

6.2. Які є відмінні риси зображення логічних функцій, притаманні таким формам: мішаній, ДФ, ДНФ, ДДНФ, МДНФ, КФ, КНФ, ДКНФ, МКНФ?

6.3. Який логічний елемент B утворюється послідовним з'єднанням (умовно позначимо його знаком $+$) елементів HI на кожному з двох входів елемента A або (та) на його виході таким чином: а) $A+HI=B$, б) $HI+A=B$, в) $HI+A+HI=B$, якщо A є елемент: 1) АБО, 2) І, 3) виняткове АБО, 4) НІ, 5) АБО-НІ, 6) І-НІ, 7) виняткове АБО-НІ, 8) імплікатор? Зобразіть 24 такі схеми.

6.4. Шляхом тотожних перетворень доведіть еквівалентність схем, поданих на рисунках: 6.9, а, б; 6.10, ж; 6.11, ж; 6.12, б; 6.13, д, е та за допомогою аналогічних перетворень наведіть паралельні схеми для елемента виняткове АБО-НІ.

6.5. Доведіть співвідношення:

1) $\overline{8 \dots 12}$ з табл. 6.2;

2) $\overline{x_1 + x_1 x_2} = \overline{x_1} + x_2$;

3) $\overline{x_1 \oplus x_2} = \overline{x_1} \oplus \overline{x_2} = \overline{x_1 \oplus x_2}$;

4) $\overline{x_1} \oplus \overline{x_2} = x_1 \oplus x_2$;

5) $\overline{x_1 \oplus x_2 \oplus x_3} = \overline{x_1} \oplus \overline{x_2} \oplus \overline{x_3} = \overline{x_1 \oplus x_2 \oplus x_3}$;

6) $\overline{x_1 \oplus x_2 \oplus x_3} = \overline{x_1} \oplus \overline{x_2} \oplus \overline{x_3} = \overline{x_1 \oplus x_2 \oplus x_3}$;

7) $\overline{y_1 \oplus y_2} = \overline{(y_1 + y_2)} \setminus y_1 y_2$;

8) $\overline{y_1 + y_2} = \overline{y_1 y_2} \setminus (y_1 + y_2)$;

9) $y_1 \setminus (y_1 \setminus y_2) = y_2 \setminus (y_2 \setminus y_1)$;

10) $y_2 \setminus (y_1 y_2) = (y_1 + y_2) \setminus y_1$;

11) $y_1 + (y_2 \setminus y_1) = y_1 + y_2$;

12) $y_1 y_2 + (y_2 \setminus y_1) = (y_1 + y_2) \setminus (y_1 \setminus y_2)$;

- 13) $y_1 y_2 \subset y_1 + y_2$;
- 14) $y_1 \subset y_2$, якщо $y_1 y_2 = y_1$ або $y_1 + y_2 = y_2$ або $\overline{y_1 y_2} = 0$;
- 15) $y_3 \subset y_1$ та $y_3 \subset y_2$, якщо $y_1 y_2 = y_3$;
- 16) $y_1 = \overline{y_2}$, якщо $y_1 + y_2 = 1$ та $y_1 y_2 = 0$;
- 17) $\overline{y_2} = y_1 \oplus y_1 y_2$, якщо $y_1 + y_2 = 1$ та $y_1 y_2 \neq 0$;
- 18) $y_2 = y_1 \oplus (y_1 + y_2)$, якщо $y_1 y_2 = 0$;
- 19) $y_1 \setminus y_2 = y_1$ та $y_2 \setminus y_1 = y_2$, якщо $y_1 y_2 = 0$;
- 20) $y_2 = y_1 \oplus (y_1 \setminus y_2)$, якщо $y_2 \subset y_1$.

Співвідношення 7...20 проілюструйте графічно діаграмами Венна.

6.6. Спростіть вирази:

- 1) $(\overline{x_1} + \overline{x_2})(\overline{x_1} + \overline{x_3})(\overline{x_2} + \overline{x_3})$;
- 2) $x_1 \overline{x_2} + x_1 \overline{x_3} + x_1 x_2 + x_2 \overline{x_3}$;
- 3) $(\overline{x_1} + x_2)(\overline{x_1} + \overline{x_3})(\overline{x_2} + \overline{x_3})$;
- 4) $(x_1 + \overline{x_2})(\overline{x_1} + x_2)(\overline{x_1} + \overline{x_3})(\overline{x_2} + \overline{x_3})$;
- 5) $(\overline{x_1} + \overline{x_2} + \overline{x_3})(\overline{x_1} + x_2 + \overline{x_3})(\overline{x_1} + \overline{x_2} + x_3 + \overline{x_4})$;
- 6) $x_1 \overline{x_2} + x_1 \overline{x_3} + x_1 \overline{x_4} + x_2 \overline{x_3} + x_2 \overline{x_4}$;
- 7) $x_1 \overline{x_2} + x_3 + x_2 \overline{x_1} + x_3 + x_3 \overline{x_1} + x_2 + x_1 + x_2 + x_3$;
- 8) $(x_1 + \overline{x_2})(\overline{x_1} + \overline{x_3})(\overline{x_1} + \overline{x_4})(\overline{x_1} + \overline{x_5})(\overline{x_2} + x_5)$.

Вказівка. За необхідністю скористайтесь діаграмами термів.

6.7. Мінімізуйте логічну функцію за методом Квайна – Мак-Класкі, яка набуває значення лог. 1 на кортежах i :

- 1) 3, 4, 5, 7, 9, 11, 12, 13;
- 2) 1, 3, 5, 7, 9, 11, 15;
- 3) 0, 2, 3, 5, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, а на всіх інших кортежах – лог. 0;
- 4) якщо на кортежах, зазначених у п. 3 цієї вправи, функція набуває значення лог. 0, а на всіх інших – лог. 1.

6.8. Для функції, заданої таким чином:

1) словесно: $y=1$, якщо $x_4 x_3 \geq x_2 x_1$, тобто якщо код, утворений двома старшими бітами вхідного кортежу, більший або дорівнює кодові з його двох молодших бітів;

2) множиною мінтермів $\bar{y} = \{M_1, M_4, M_5, M_7\}$, а $M_0, M_2, M_3, M_6 \notin \bar{y}$;

3) діаграмою термів D_{35} на рис. 6.14, б;

4) схемою по виходу y_6 на рис. 6.14, в;

5) функцією довільної форми:

$$y = (x_1 + x_2 + x_3) \overline{(x_2 + x_1 x_4)} \overline{(x_4 + x_1 x_3)} \cdot \overline{x_1 + x_2 \cdot x_3 + x_4 \cdot x_4 x_1 + x_2 + x_3};$$

6) нулями на діаграмі термів у клітинках $i=0, 1, 4, 10, 11, 14, 16, 17, 20, 26, 27, 30$ та всіма іншими одиницями;

7) одиницями в зазначених у п. 6 цієї вправи клітинках i діаграми та нулями в усіх інших;

8) $y=1$ на кортежах $i=1, 2, 5, 6, 8, 11$, $y=0$ при $i=0, 7, 9, 10, 13$ та невизначену на інших кортежах, наведіть мінімальні форми для побудови схеми в базисах: а) булевому в послідовності НІ, І, АБО; б) булевому в послідовності НІ, АБО, І; в) І-НІ; г) АБО-НІ; д) І-АБО-НІ; е) у мішаній формі, за змогою з використанням елементів виключне АБО(-НІ). Визначте складність та швидкодію схем.

6.9. Мінімізуйте схему в елементному базисі І-НІ чи АБО-НІ, який забезпечує меншу складність (на чотирьох логічних елементах за однофазного надходження змінних), для заданої логічної функції:

$$1) y = (1 \setminus x_3) \oplus (x_1 \rightarrow \bar{x}_2);$$

$$2) y = 1 \setminus [x_2 \oplus (\bar{x}_3 \rightarrow x_1)];$$

$$3) y = \overline{x_3 \setminus x_2} \rightarrow (\bar{x}_1 \oplus \bar{x}_3) + (x_2 \setminus x_1);$$

$$4) y = 1 \setminus [(x_1 \rightarrow x_2 x_3) \rightarrow (x_2 \oplus x_3)];$$

$$5) y = 1 \setminus [(x_2 + x_4) \rightarrow x_3 \overline{x_1 \oplus x_3}];$$

$$6) y = 1 \setminus [(\bar{x}_1 \rightarrow x_4) \setminus (x_3 \overline{x_2 \oplus x_3})];$$

$$7) y = 1 \setminus \left[\left(x_1 + x_2 + \overline{x_3} \right) \rightarrow \overline{x_3 x_2 \oplus x_4} \setminus \left(\overline{x_1 x_2 x_3} \right) \right];$$

$$8) y = \left(x_1 + \overline{x_2 x_3} \right) \rightarrow x_1 + \left(x_2 \oplus x_4 \right) \setminus \left(\overline{x_2 x_3} \right);$$

$$9) y = \left(x_2 + \overline{x_1 x_3 + x_4 x_5} \right) \rightarrow \left(x_4 x_5 \setminus \overline{x_1 \oplus x_2} \right);$$

$$10) y = 1 \setminus \left[\left(x_1 + x_2 + x_3 \oplus \overline{x_4 x_3} \right) \rightarrow \overline{x_4 x_3 + x_5} \right].$$

6.10. Виконайте спільну мінімізацію чотирьох вихідних функцій, що здійснюють перетворення ДДК 8421 згідно з табл. 5.6, 5.7 у такі коди: а) 2421, б) 7421, в) 8421+3 (з надлишком три), г) код Грея (перші десять цифр $X_{10}=0\dots 9$ чотирирозрядного коду). *Вказівки.* На заборонених кортежах $X_{10}>9$ вважати функції невизначеними. Схему мінімальної складності реалізуйте на довільних логічних елементах.

7. БАЗОВІ ЕЛЕМЕНТИ ЦИФРОВИХ ІНТЕГРОВАНИХ МІКРОСХЕМ

Логічний елемент, який здійснює одну й ту саму функцію, можна побудувати на основі електронних ключів різного типу, що відрізняються схемним і технологічним виконанням. *Базовий елемент* (БЕ) цифрових інтегрованих мікросхем (ІС) є логічний елемент конкретного типу технології та схемотехніки. Серед сучасних цифрових ІС набули поширення БЕ на основі біполярної технології: ТТЛ – транзисторно-транзисторної логіки, ЕСЛ – емітерно-сполученої логіки, І²Л – інтегрованої інжекційної логіки, а також на основі уніполярної технології: МОНТЛ – на МОН-структурах та КМОНТЛ – на комплементарних МОН-структурах.

Комплект ІС різного призначення, побудованих на основі БЕ одного типу, що має єдине конструктивно-технологічне виконання, утворює *серію* ІС. На основі ІС однієї серії, а іноді також різних серій з однаковим конструктивним виконанням зручно будувати пристрої, тому що не має потреби в електричному та конструктивному узгодженні елементів.

7.1. Базові елементи ТТЛ

7.1.1. Базові елементи з простим інвертором

До комплекту серій ІС ТТЛ входить понад 100 типоміналів – різновидів ІС за функційним призначенням. Поширення ІС ТТЛ зумовлене їх універсальністю, добре розвиненою технологією і схемотехнікою, а також параметрами, що задовольняють вимоги побудови пристроїв різного призначення.

Елементи ТТЛ походять від схем діодно-транзисторної логіки (ДТЛ). БЕ ДТЛ (рис. 7.1, *a*) виконує логічну функцію І-НІ та утворюється від елемента І діодної логіки додаванням на його виході ключа-інвертора на транзисторі VT, емітерний перехід $V_{\text{бе}}$ якого еквівалентний діоду $VD_{\text{н}}$.

З надходженням хоча б на один з входів x_1, x_2 напруги рівня лог. 0 відповідний вхідний діод відчиняється, тому напруга в точці *A* схеми стає низькою, недостатньою для відкривання

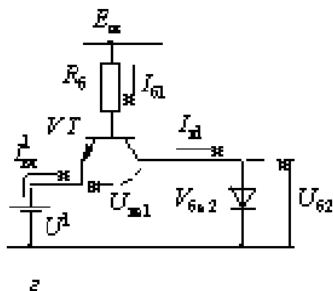
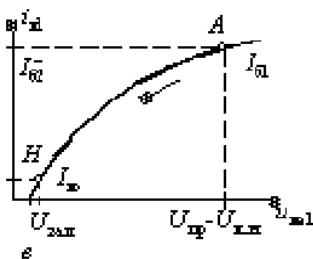
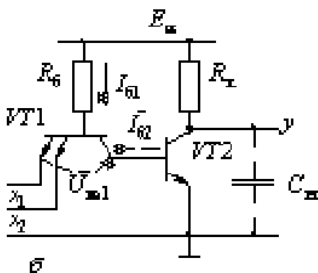
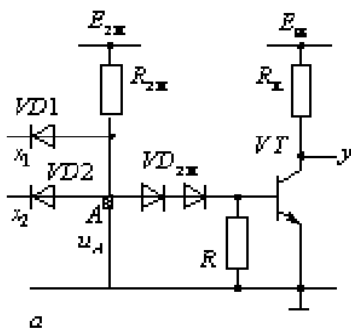


Рис. 7.1. Базовий елемент ТТЛ

трьох $p-n$ -переходів: двох діодів зміщення $VD_{зм}$ та емітерного переходу $V_{бе}$, внаслідок чого транзистор VT опиняється в режимі відтину й на його виході y встановлюється напруга

рівня лог. 1: $U^1 \approx E_{ж}$. Лише з надходженням до всіх входів напруг рівня лог.1 вхідні діоди зачиняються, струм від джерела $E_{зм}$ перемикається до бази транзистора, тому останній переходить до режиму насичення і напруга на його виході y зменшується до рівня лог. 0:

$U^0 = U_{к.н}$. Через те, що входи x_i виявляються при цьому ізольованими, елемент не споживає струм від джерела сигналу.

БЕ ТТЛ (рис. 7.1, б) відрізняється від БЕ ДТЛ тим, що в ньому роль вхідних діодів, які виконують функцію І, та діодів зміщення відіграє багате мітерний транзистор (БЕТ) $VT1$: його емітерні переходи $V_{бе1}$ еквівалентні діодам $VD1, VD2$, а колекторний перехід $V_{бк1}$ – діоду $VD_{зм}$. Ключ-інвертор на транзисторі $VT2$, як і раніше, забезпечує відновлення стандартних логічних рівнів на виході схеми. Крім того, БЕ пришвидшує процес вимикання цього ключа.

Формально принцип

дії елементів аналогічний. Проте внаслідок взаємодії переходів у БЕТ є відмінність між елементами ТТЛ і ДТЛ. Під час зменшення вхідної напруги, наприклад, на вході x_1 до рівня лог. 0 зворотний струм $I_{б2}$ розсмоктування заряду в насиченому транзисторі $VT2$ замикається через транзистор $VT1$, який при цьому опиняється в активному режимі: його робоча точка A (рис. 7.1,

в) визначається струмом $I_{к1} = I_{б-}$ значної величини і напругою $u_{ке1} = U_{пр} - U_{к.н}$ (при з'єднанні входу x_1 з колектором насиченого

транзистора). Через це струм $I_{б-} = h_{21e} I_{б1}$ підсилюється транзистором $VT1$ і пришвидшує процес розсмоктування та вимикання транзистора $VT2$ (у БЕ ДТЛ розсмоктування затягується, тому що струм $I_{б}$ протікає через резистор R з великим опором). В міру зачинення ключа $VT2$ робоча точка рухається траєкторією $A-H$ і, отже, БЕТ переходить до режиму насичення з мікрострумом в колі колектора, що дорівнює зворотному тепловому струму $I_{ко}$ транзистора $VT2$. На початку координат точці H відповідає залишкова напруга $u_{ке1} = U_{зал} < U_{к.н}$, яка за величиною не перевищує $0,05 \dots 0,1 B$, тому напруга на базі транзистора $VT2$ близька до вхідної: $u_{б2} = U^0 + U_{зал} < U_{б0}$ і підтримує його в стані відтину.

З надходженням на всі входи напруги рівня лог. 1 всі емітерні переходи БЕТ $VT1$ зачинаються, а його колекторний перехід $V_{бк1}$ та базовий перехід $V_{бе2}$ інвертора відчинаються і транзистор $VT2$ переходить до режиму насичення. Елементарний транзистор VT у складі БЕТ (рис. 7.1, з) при цьому виявляється в активному інверсному режимі – ролі емітера та колектора міняються місцями, тому струм $I_{к1} = I_{е1} + I_{б1} \approx I_{б1}$, бо інверсний емітерний струм, що втікає в схему $I_{вх}^I = I_{е1} = h_{21e.1} I_{б1} \ll I_{б1}$ через те, що в інверсному ввімкненні коефіцієнт $h_{21e.1} \ll 1$. Отже, від джерела сигналу до емітера втікає вхідний струм $I_{вх}^I$, що навантажує його, тому цей струм зменшують до мінімуму; типове значення $I_{вх}^I \leq 40 \text{ мкА}$.

Позитивною якістю розгляненого елементу ТТЛ є підвищення його швидкодії відносно ДТЛ внаслідок того, що пришвидшується розсмоктування заряду в інверторі, а недоліком є зниження завадостійкості, бо в колі зміщення ТТЛ на один p -перехід менше. Крім того, як і в БЕ ДТЛ та в звичайному ключі, в елементі ТТЛ затягується фронт вимикання під час заряджання ємності навантаження C_H через великий опір резистора R_K . Ці недоліки усуваються в елементах зі складним інвертором, які й набули практичного застосування.

7.1.2. Основна схема та статичні режими

Базова схема елемента ТТЛ (на рис. 7.2, а виділена пунктиром) складається з вхідного каскаду на БЕТ VT_1 , який виконує, як і в попередній схемі, логічну функцію І, фазорозділювального каскаду VT_2 , що виконує функцію комутатора, та вихідного двотактного каскаду на транзисторах VT_3 , VT_4 .

Розглянемо для спрощення статичні стани БЕ в режимі інвертора: до входу $x_2=1$ прикладено сталу напругу рівня лог. 1, тому відповідний емітерний перехід зачинений і не впливає помітно на роботу елемента. Напругою u_{x1} на вході x_1 , з'єднаного з виходом такого самого БЕ на транзисторах VT_3 , VT_4 , струм I_{B1} від джерела живлення $E_{жс}$ перемикається то в емітер БЕТ через перехід V_{be1} , зв'язаний зі входом x_1 , то в емітерні переходи V_{be2} , V_{be4} транзисторів VT_2 , VT_4 через колекторний перехід V_{bc1} БЕТ. Умовно така комутація зображена за допомогою перемикача $S1$ на рис. 7.2, б. Схема на транзисторах VT_2 , VT_3 , VT_4 є складним інвертором: фазорозділювальний каскад VT_2 комутує транзистори вихідного каскаду, підмикаючи по черзі вихід у до високої напруги U^1 або до низької U^0 , як це умовно показано на рис. 7.2, в за допомогою двопозиційного перемикача $S2$. При цьому вихід у навантажено на кількість входів $K_{роз}$ аналогічних БЕ ($K_{роз}$ – коефіцієнт розгалуження або навантажувальна здатність). Завдяки двотактному каскаду забезпечується малий вихідний опір в обох станах перемикача, внаслідок чого підвищу-

ється навантажувальна здатність і швидкодія БЕ ТТЛ.

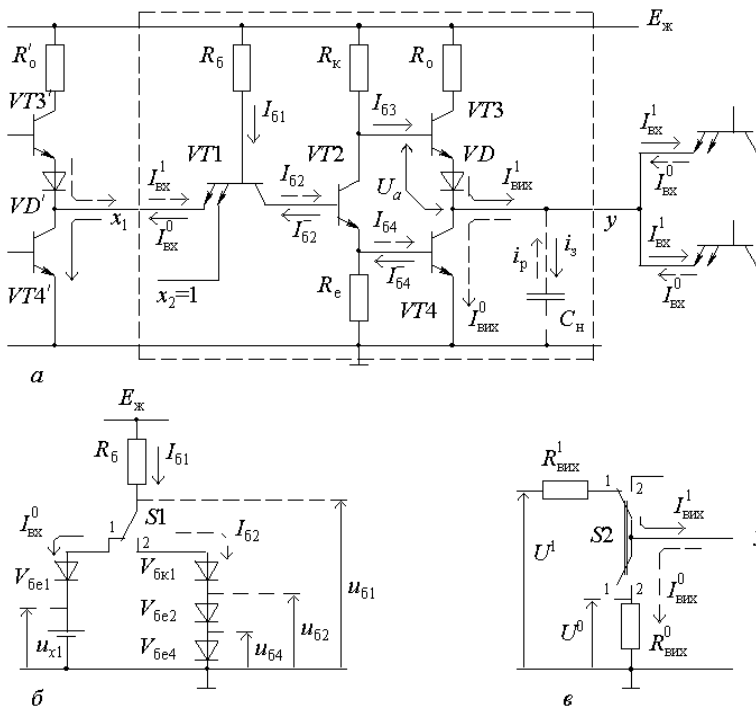


Рис. 7.2. Основна схема базового елемента ТТЛ

У **вимкненому стані** рівнем $x_1=0$ на виході насиченого транзистора-джерела $VT4'$ мітерний перехід V_{be1} БЕТ $VT1$ відчинено (перемикач $S1$ в положенні 1) і через нього з елемента витікає вхідний струм $I_{вх}^0$ значної величини (у цьому стані струми подано суцільними стрілками), який замикається через транзистор $VT4'$. Напруга низького рівня

$$u_{б1} = U_{x1} + U_{be1} = U_{к.н} + U_{пр} < 3U_{б0} \quad (7.1)$$

недостатня для відкриття трьох $p-n$ -переходів транзисторів $VT1$, $VT2$, $VT4$ (див. рис. 7.2, б), тому транзистор $VT2$ зачинений і до колектора БЕТ $VT1$ замикається лише його тепловий струм $I_{к1} = I_{б2} = I_{к0}$. Внаслідок цього транзистор $VT1$ опиняється в ре-

жимі насичення з мікрострумом в його колекторному колі, тому напруги становлять

$$u_{\text{б2}}=u_{\text{x1}}+u_{\text{ке1}}=U_{\text{к.н}}+U_{\text{зал}} \approx U_{\text{к.н}}, \quad u_{\text{бк1}}=u_{\text{б1}}-u_{\text{б2}} \approx U_{\text{пр}}. \quad (7.2)$$

Отже, обидва переходи транзистора $VT1$ зміщено в прямому напрямку, що пришвидшує процес розсмоктування заряду в транзисторі $VT2$ під час його закривання.

Зачинений транзистор $VT2$ комутує вихідний каскад таким чином: напругою $u_{\text{е2}}=u_{\text{б4}}=I_{\text{е2}}R_{\text{е}} \approx 0$ транзистор $VT4$ вимикається, а напругою $u_{\text{к2}} \approx E_{\text{ж}}$ вмикаються транзистор $VT3$ і діод VD , що еквівалентно стану 1 двопозиційного перемикача $S2$. Тому на виході у встановлюється напруга рівня лог. 1:

$$u_{\text{у}}=U^1=u_{\text{к2}} - u_{\text{бє3}} - u_{\text{VD}} \approx E_{\text{ж}} - U_{\text{пр}}, \quad (7.3)$$

яка за типового значення $E_{\text{ж}}=5 \text{ В}$ становить $U^1 \approx 3,5 \text{ В}$. Наскрізний струм через вихідний каскад при цьому не протікає, а вихідний струм $I_{\text{вих}}^1$ витікає з схеми і розгалужується до входів навантажувальних елементів, отже, струм $I_{\text{вих}}^1=K_{\text{роз}}I_{\text{вх}}^1$ невеликий, тому що $I_{\text{вх}}^1 \leq 40 \text{ мкА}$. За відсутності навантаження вихідний струм дорівнює лише тепловому струму транзистора $VT4$.

Резистор R_0 у вихідному каскаді з опором близько $0,1R_{\text{к}}$ призначений для обмеження струму короткого замикання, який може виникнути під час випадкового закорочення виходу схеми на землю, під час заряду ємності $C_{\text{н}}$ значної величини, а також в процесі перемикавання вихідного каскаду, коли обидва транзистори $VT3$ та $VT4$ можуть опинитися у відчиненому стані. Навантаження ввімкнене в емітерне коло транзистора $VT3$, тобто цей каскад є емітерним повторювачем. Через малі значення опору R_0 і струму $I_{\text{вих}}^1$ напруга на колекторі $u_{\text{к3}} \approx E_{\text{ж}}$, а на базі $u_{\text{б3}} \leq E_{\text{ж}}$, отже, колекторний перехід зачинений, тому транзистор $VT3$ перебуває в активному режимі. Цим пояснюється невеликий вихідний опір БЕ у вимкненому стані: $R_{\text{вих}}^1 \approx R_{\text{к}}/h_{21\text{е}}$.

Таким чином, при $x_1 x_2 = 0$ БЕ вимкнено, транзистори перебувають у таких станах: $VT1$ – в режимі насичення, $VT2$, $VT4$ – в режимі відтину, $VT3$ – в активному режимі, діод VD – відчинений; вхідний і вихідний струми витікають із БЕ; напруги: на вході $U^0 \approx 0,2 \text{ В}$ та на виході $U^1 \approx 3,5 \text{ В}$.

Статичний **увімкнений стан** забезпечується рівнем $x_1 = 1$ на виході відчиненого верхнього плеча $VT3'$, VD' ерувального каскаду. При цьому емітерний перехід V_{be1} БЕТ $VT1$ зміщується у зворотному напрямку, а три p - n -переходи транзисторів $VT1$, $VT2$, $VT4$ через резистор R_6 підмикаються до джерела живлення $E_{ж}$ і тому виявляються зміщеними в прямому напрямку (перемикач $S1$ у положенні 2). БЕТ переходить до активного інверсного режиму, а транзистори $VT2$ та $VT4$ – до режиму насичення. До БЕ втікає малий струм $I_{вх}^1$ (у цьому стані елемента струми зазначено пунктиром), струм I_{61} втікає до бази транзистора $VT2$, а емітерний струм останнього частково замикається до бази транзистора $VT4$.

До пари p - n -переходів верхнього плеча вихідного каскаду V_{be3} транзистора $VT3$ та діода VD прикладено напругу $u_a = u_{к2} - u_{к4} = U_{пр} < 2U_{60}$, якої не вистачає для їх відкривання, тому транзистор $VT3$ та діод VD зачиняються і, отже, цей діод виконує функцію діода зміщення. Еквівалентний перемикач $S2$ виявляється в положенні 2. Вихідний опір $R_{вих}^0 = R_{вих.н}$ малий і становить кілька одиниць Ом, а напруга $u_y = U^0 = U_{к.н}$ відповідає рівню лог. 0. Через вихідний каскад наскрізний струм відсутній, а до нижнього плеча втікає значний вихідний струм $I_{вих}^0 = K_{роз} I_{вх}^0$ від елементів-навантажень.

Отже, при $x_1 x_2 = 1$ БЕ ввімкнено, БЕТ $VT1$ перебуває в активному інверсному режимі, транзистори $VT2$, $VT4$ – в насиченому режимі, транзистор $VT3$ та діод VD зачинені, вхідний і вихідний струми втікають до БЕ, напруги: на вході $U^1 \approx 3,5 \text{ В}$ та на виході $U^0 \approx 0,4 \text{ В}$.

7.1.3. Статичні характеристики

Для правильного використання ІС та їх можливостей, особливо при сполученні з іншими елементами (дискретними або ІС інших серій) необхідно представляти, в яких межах зміни струмів та напруг ІС може нормально функціонувати. З цією метою розглядаються статичні характеристики БЕ: вхідна $i_{\text{вх}}(u_x)$, передатна $u_y(u_x)$ та вихідні $i_{\text{вих}}(u_y)$, що знімаються за повільної зміни напруг і струмів.

На *вхідній* характеристиці (рис. 7.3, *а*) вимкненому стану БЕ відповідає зображувальна точка 1; на ділянці 1-0 вхідний струм $I_{\text{вх}}^0$, що витікає з елемента, збільшується під час зменшення напруги u_x , у тому числі до від'є'них значень і при вхідній напрузі, більшій за величиною, ніж допустима $U_{\text{доп}}$, різко зростає, внаслідок чого в БЕТ може настати тепловий пробій. Для обмеження негативної напруги на входах усіх сучасних серій ТТЛ вбудовують діоди (фрагмент схеми подано на рис. 7.4, *а*), які називають також демпферними, тому що вони сприяють затуханню паразитних коливань, які виникають в лініях зв'язку через відбиття сигналів від їх кінців між керувальними та навантажувальними елементами.

Зі збільшенням вхідної напруги на ділянці 1-3 струм $I_{\text{вх}}^0$ зменшується і під час переходу її через пороговий рівень (ділянка 3-4) на вході встановлюється рівень лог. 1, БЕ вмикається і вхідний струм $I_{\text{вх}}^1 \leq 40 \text{ мкА}$ (ділянка 4-7) тепер втікає до елемента. При перевищенні вхідною напругою рівня $U_{\text{доп}}^+ \geq 5,5 \text{ В}$ може виникнути електричний пробій емітерного переходу БЕТ (точка 7). Усунути пробій, якщо може виникнути його небезпека, можна за допомогою обмежувальних діодів, як подано на фрагменті схеми рис.7.4, *б*.

На ділянці 0-3 вхідний опір БЕ визначається резистором у колі бази БЕТ і становить $R_{\text{вх}}^0 \approx R_6 \approx 4 \text{ кОм}$, тому що БЕТ насичений, а транзистор VT2 зачинений; у навколорогової зоні 3-4 через відкривання транзистора VT2 вхідний опір зменшується до

$R_{\text{вх}}^0 \approx (300 \dots 400) \text{ Ом}$; і, нарешті, на ділянці 4-7 емітерні переходи БЕТ зачиняються, тому опір зростає до $R_{\text{вх}}^1 \approx 2 \text{ МОм}$.

На *передатній* характеристиці (рис. 7.3, б) точка 1 відповідає статичному вимкненому стану. Зі збільшенням вхідної напруги відповідно з (7.1) зростає також напруга на базі транзистора VT2 і, коли вона сягає $u_{\text{б2}} = U_{\text{б0}}$ (точка 2), транзистор VT2 відчиняється: напруга на його емітері $u_{\text{е2}} = u_{\text{б4}} = I_{\text{е2}} R_{\text{е}}$ зростає, а на колекторі $u_{\text{к2}} = E_{\text{ж}} - I_{\text{к2}} R_{\text{к}}$ зменшується, внаслідок чого приріст вхідної напруги Du_{x} потрапляє на вихід елементу (ділянка 2-3): $u_{\text{y}} = u_{\text{к2}} - U_{\text{пр}}$, що призводить до погіршення його завадостійкості.

Для усунення цього недоліку резистор $R_{\text{е}}$ замінюють коригувальним каскадом на транзисторі VT5 (рис. 7.4, в). Тепер зі збільшенням вхідної напруги до величини приблизно $2U_{\text{б0}}$ (точка 6) транзистор VT2 залишається зачиненим, тому що між його базою та корпусом увімкнено два *p-n*-переходи транзисторів VT2 та VT4 або VT2 та VT5, тобто похила ділянка 2-3 замінюється лінією 2-6. Завдяки поліпшенню прямокутності передатної характеристики статична завадостійкість збільшується майже вдвічі. Коли, в міру зростання, напруги сягають величини $u_{\text{б2}} = 2U_{\text{б0}}$ та $u_{\text{е2}} = u_{\text{б4}} = U_{\text{б0}}$ (точки 6 або 3), починається вмикання БЕ: відчиняється транзистор VT4, вхідний опір якого зменшується, тому до його бази відгалужується значна частина струму $I_{\text{е2}}$, що спричиняє швидкий перехід пари VT2, VT4 до насичення і збільшення крутизни характеристики на ділянці 6-3-4. Після вмикання (ділянка 4-5) транзистор VT3 виявляється в режимі відсікання, а транзистори VT2, VT4 та VT5 – в режимі насичення.

Ширина навколопорогової ділянки 6-3-4 незначна, тому *порогову напругу* $U_{\text{п}}$ вмикання та вимикання вважають однаковою і визначають за перетином характеристики з лінією одного підсилення $u_{\text{y}} = u_{\text{x}}$. Відносно порогової напруги відлічують статичну завадостійкість як рівень завад $U_{\text{з}}^0 = U_{\text{п}} - U^0$, $U_{\text{з}}^1 = U^1 -$

– U_{Π} . Для БЕ ТТЛ типовою є величина $U_{\Pi} \gg 1,5 B$, а допустимий рівень завад з урахуванням розкиду параметрів, нестабільності їх у температурному діапазоні, а також нестабільності напруги джерела живлення при паспортній навантажувальній здатності становить $U_3 \geq 0,4 B$. Вважаючи логічний перепад напруги на виході в найгіршому випадку не нижчим за $U_{\Pi} = U^1_{\min} - U^0_{\max} = 2,4 - 0,4 = 2 B$, маємо відносну завадостійкість не меншу за $U_3 / U_{\Pi} = 20\%$.

Вимкнений стан БЕ описується вихідною характеристикою А (рис. 7.3, в); зображувальній точці 1 відповідає напруга $U^1 \approx 3,5 B$ і струм, що витікає з виходу, $I^1_{\text{вих}} = K_{\text{роз}} I^1_{\text{вх}} \leq 0,4 \text{ мА}$ при $K_{\text{роз}} \leq 10$. На робочій ділянці завширшки $U^1 = (2,4 \dots 4) B$ емітерний повторювач VT3 перебуває в активному стані. Якщо вихід схеми підімкнути до високої напруги $U^1 \geq 4 B$, транзистор VT3 та діод VD зачинаються і до вихідного кола починає замикатися малий струм втрати (ділянка В), а за великої напруги $U^1 \approx 10 B$ може настати електричний пробій колекторного переходу транзистора VT4 (ділянка С). Зі збільшенням навантаження зростає величина струму $I^1_{\text{вих}}$ та спаду напруги на резисторі R_0 , внаслідок чого зменшується вихідна напруга. При $U^1 \leq 2,4 B$ відчиняється колекторний перехід транзистора VT3 і він потрапляє до режиму насичення, вихідний опір стає близько $R^1_{\text{вих}} \approx R_0$. При закороченні виходу струм $I^1_{\text{вих}}$ збільшується до $(30 \div 40) \text{ мА}$, а під час підімкнення до виходу негативної напруги цей струм різко зростає (ділянка D) і вихідна напруга обмежується на рівні $-(0,8 \dots 1) B$ відчиненим паразитним діодом $V_{\text{ПК}}$ між підшарком ("з'млею") та колектором транзистора VT4. Отже, БЕ нормально функціонує на ділянці А завширшки U^1 .

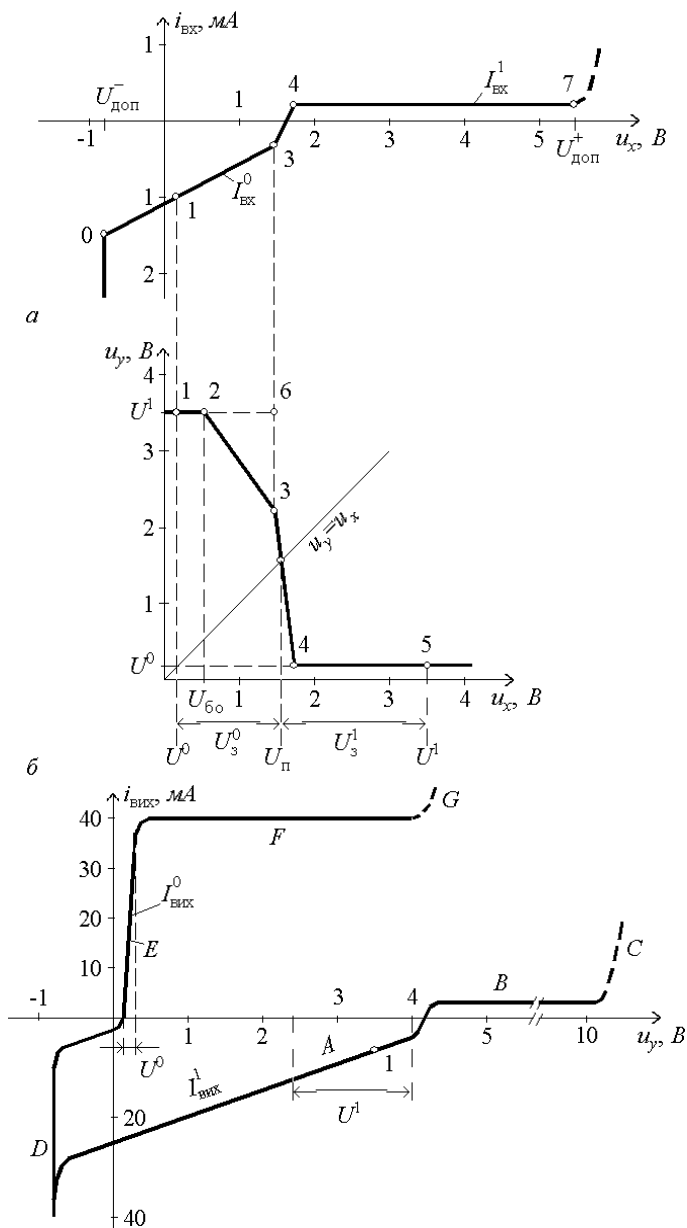


Рис. 7.3. Вхідна (а), передатна (б) і вихідна (в) характеристики БЕ ТТЛ

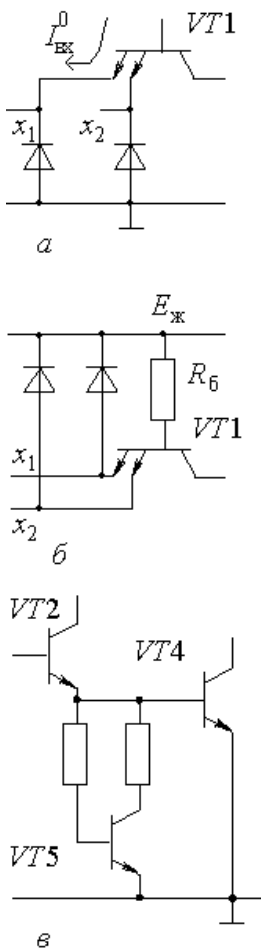


Рис. 7.4. Схеми модифікації БЕ ТТЛ

Увімкненому стану БЕ відповідає вихідна характеристика E : на робочій ділянці завширшки $U^0 = (0,2 \dots 0,4) \text{ В}$ зі збільшенням вихідного струму $I_{\text{вих}}^0 = K_{\text{роз}} I_{\text{вх}}^0$, що втікає до насиченого транзистора $VT4$, зростає спад напруги на вихідному опорі $R_{\text{вих}}^0 = R_{\text{вих.н}} \approx 10 \text{ Ом}$ до припустимого рівня $U^0 = 0,4 \text{ В}$. За подальшого збільшення вихідної напруги (ділянка F) транзистор $VT4$ переходить до активного режиму і струм стабілізується, доки не настане тепловий пробій у цьому транзисторі (ділянка G) при $U_y > 4 \text{ В}$.

З надходженням до виходу негативної напруги транзистор $VT4$ переходить до активного інверсного режиму, струм змінює напрямок і витікає з транзистора, а з відкриванням зазначеного діода $V_{\text{пк}}$ (ділянка D) напруга на виході обмежується величиною $-(0,8 \dots 1) \text{ В}$. Отже, у увімкненому стані БЕ нормально функціонує на ділянці E завширшки U^0 . Під час зміни полярності сигналу, що надходить до виходу БЕ, зокрема, через відбиття від кінців лінії зв'язку, струм через БЕ може протікати в обох напрямках, що сприяє виникненню паразитних коливань у лініях, які зв'язують БЕ з його елементами-навантаженнями.

7.1.4. Перехідні процеси

Характер перехідних процесів у БЕ ТТЛ, як і у звичайних ключах, зумовлений інерційністю транзисторів, розсмоктуванням нагромадженого в них заряду та перезарядженням пара-

зитних ємностей. Нехай у початковому стані БЕ вимкнений напругою лог. 0 на вході x_1 , а рівень $x_2=1=\text{const}$ (див. рис. 7.2, а).

Вмикання відбувається при збільшенні вхідної напруги до порогового рівня $u_x=U_{\text{п}}$, з моменту t_1 (рис. 7.5). При цьому струм $I_{\text{б1}}$ перемикається спочатку на заряджання паразитних ємностей на входах транзисторів VT2, VT4, а відтак – у їх бази, спричиняючи насичення цих транзисторів. Вхідна ємність транзистора VT3 розряджається через транзистор VT2, внаслідок чого напруга u_a зменшується і елементи VT3, VD зачинаються.

До перемикавання вихідного каскаду напруга u_y майже не змінюється, бо навантажувальна ємність $C_{\text{н}}$ лишається зарядженою, а після насичення транзистора VT4 відбувається її швидкий розряд через малий опір транзистора і формування фронту спаду напруги $t_{\text{ф}}^{10}$. Під час перемикавання струм живлення $i_{\text{ж}}$ збільшується від $I_{\text{ж}}^1$ до $I_{\text{ж}}^0$ і витрачається, в основному, на підтримання транзисторів VT2, VT4 в режимі насичення.

Завдяки використанню високочастотних транзисторів, заряду ємностей і відкриванню транзисторів VT2, VT4 інтенсивним струмом $I_{\text{б}}^1$, відсутності етапу розсмоктування під час закривання транзистора VT3 (тому що в увімкненому стані БЕ він перебуває в активному режимі), а також малій сталій часу розряду навантажувальної ємності, процес вмикання відбувається швидко. Час затримки поширення під час вмикання для типових елементів ТТЛ не перевищує $t_{\text{з.п}}^{10} \leq 15 \text{ нс}$.

Вимикання здійснюється негативним перепадом сигналу на вході x_1 . Починаючи з моменту t_2 , коли напруга u_x сягає порогового рівня $U_{\text{п}}$, відчиняється емітерний перехід транзистора VT1, БЕТ переходить до активного режиму, а з часом – і до режиму насичення. Завдяки цьому утворюється низькоопірний шлях зворотного струму $I_{\text{б4}}, I_{\text{б2}}, I_{\text{вх}}^0$ (як показано суцільними стрілками на рис. 7.2, а), що сприяє швидкому розсмоктуванню заряду і зачиненню спочатку транзистора VT2, а згодом – і VT4.

Закривання транзистора VT_2 спричиняє зростання колекторної напруги $u_{к2}$ і відкриття транзистора VT_3 та діода VD раніше, ніж встигне закритися транзистор VT_4 . Це зумовлює сплеск наскрізного струму $i_{ж}$ через вихідний каскад, що може привести до виникнення завади в шині живлення.

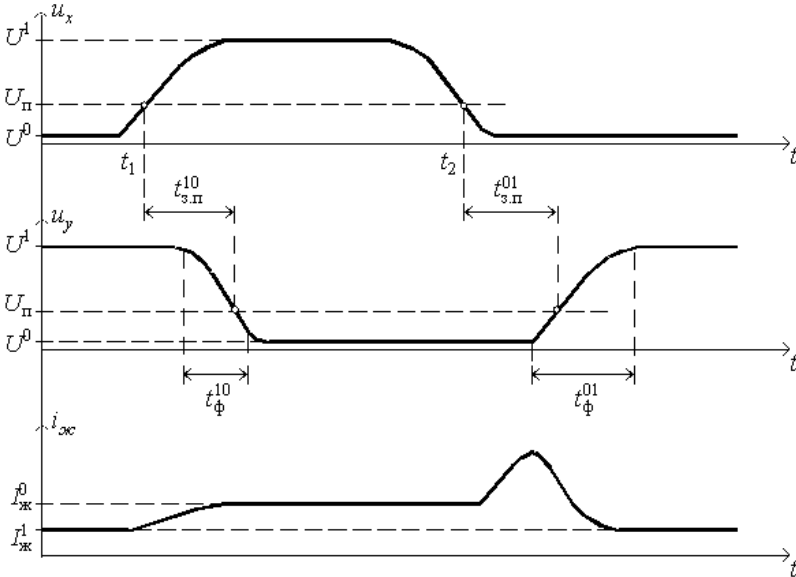


Рис. 7.5. Часові характеристики базового елемента ТТЛ

При закритті транзистора VT_4 навантажувальна ємність C_H заряджається через вихідний опір емітерного повторювача $R_{вих}^1$, зумовлюючи фронт зростання $t_{ф}^{01}$ напруги до рівня U^1 . Затримка вимикання, спричинена розсмоктуванням надлишкового заряду та зарядом ємності C_H через опір $R_{вих}^1 > R_{вих}^0$, більша, ніж $t_{з.п}^{10}$ і для типових елементів ТТЛ не перевищує $t_{з.п}^{01} \leq 22 \text{ нс}$.

7.1.5. Модифікації базового елемента

З метою поліпшення параметрів БЕ, полегшення сполу-

чення їх з іншими пристроями, розширення функціональних можливостей поряд з універсальними БЕ засвоєно випуск також їх різновидів.

1. Різновиди БЕ за виконуваною логічною функцією.

Для розширення функціональних можливостей у різних серіях БЕ ТТЛ передбачено логічні розширювачі щодо АБО на транзисторах $VT1'$, $VT2'$ вільними виходами k , e (рис. 7.6, а). Підімкнення їх до однойменних розширювальних входів k , e елементу І-НІ (показано пунктиром) дозволяє реалізувати функцію другого ступеня АБО, а загалом відносно входів БЕТ – функцію І-АБО-НІ. До складу серій ТТЛ входять також елементи І-АБО-НІ, в яких таке з'єднання виконано всередині ІС.

Якщо замість БЕТ у ТТЛ виконують транзистор з одним емітером, в одному корпусі ІС отримують набір елементів НІ, а з'єднання таких елементів на зразок рис. 7.6, а, утворює елемент АБО-НІ. Для отримання логічного елементу І (фрагмент на рис. 7.6, б) між БЕТ $VT1$ і фазорозділювальним каскадом $VT2$ вбудовують додатковий інвертор $VT6$, який може бути виконаний також на складеному транзисторі. Паралельне з'єднання таких вхідних кіл з транзистором $VT2$ на спільні навантаження R_k , R_e , аналогічне рис. 7.6, а, утворює елемент І-АБО.

За допомогою модифікації, яка набула назву транзистор-транзисторно-транзисторної логіки (Т-ТТЛ), складніші функції можна здійснити на меншій кількості елементів. Так, елемент заборони (елемент І з інверсним входом) реалізується схемою ТТЛ з додатковим транзистором $VT0$ (рис. 4.6, в). На вході z БЕТ $VT1$ з'являється рівень лог. 0 (рис. 7.6, г) лише при відкриванні транзистора $VT0$ сигналами $x_1=1$, $x_2=0$, а за їх інших комбінацій цей транзистор залишається зачиненим, що еквівалентно рівню $z=1$. Отже, за сталого рівня $x_3=1$ на виході елементу реалізується функція $y=x_1x_2$. Наявність іверсного входу в елементі І спрощує побудову цифрових пристроїв на електрично сумісних схемах ТТЛ і Т-ТТЛ.

2. Елементи з вільним колектором. Двотактний вихідний каскад БЕ ТТЛ забезпечує підвищення швидкодії та навантажувальної здатності, проте стає на перешкоді сполученню між

елементами через те, що *виходи кількох БЕ не можна об'єднувати* на спільне навантаження або лінію зв'язку, тому що при цьому може виникнути наскрізний струм. Наприклад, за ввімкненого стану БЕ з вихідним каскадом $VT3$, $VT4$ та вимкненого стану БЕ з виходом $VT3'$, $VT4'$ (рис. 7.7, а) крізний струм $i_{ж}$ протікає через транзистори $VT3'$ та $VT4$. Цей струм може перевищити допустиму в статичному стані величину і призвести до виходу з ладу елементів. Крім того, в лінії y встановлюється рівень напруги, проміжний між лог. 0 та лог. 1, тобто сполучені таким чином елементи не функціонують нормально.

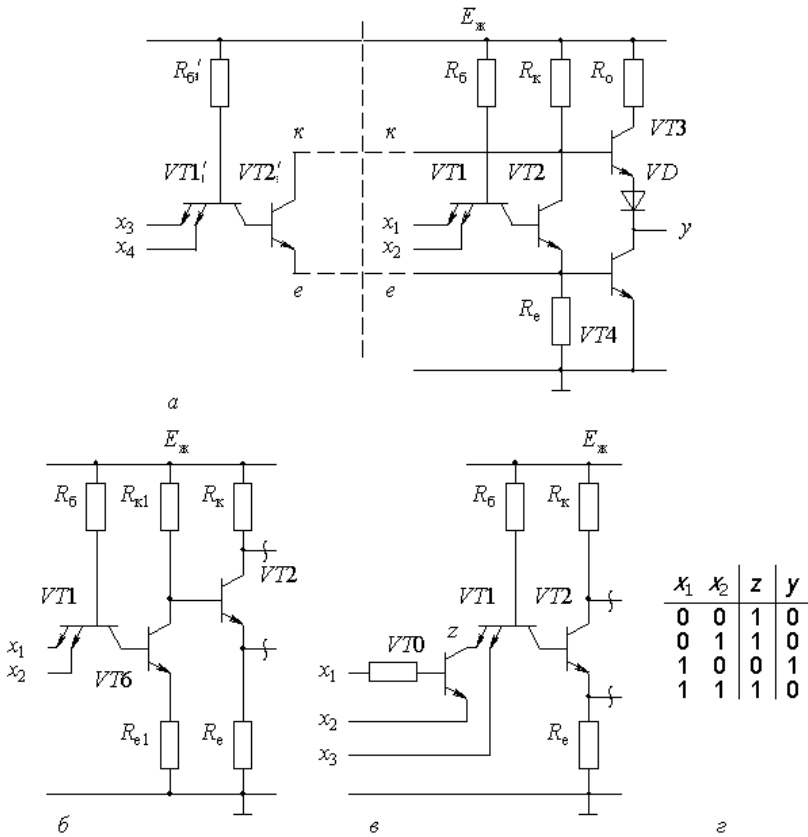


Рис. 7.6. Використання БЕ ТТЛ з відкритим колектором

Об'єднувати виходи та підмикати їх до нестандартного навантаження можна в елементах з вільним (відкритим) колектором (рис. 7.7, б), які використовуються також для керування індикаторними й іншими виконавчими приладами. На рис. 7.7, в, г наведено приклади схем індикації зі світлодіодом або лампою розжарювання, в яких світіння відбувається при $y=0$ та $y=1$ відповідно. За індуктивного навантаження, наприклад, на обмотку реле K (рис. 7.7, д) колектор вихідного транзистора захищають від електричного пробоя напругою самоіндукції за допомогою демпферного діода VD . Для керування потужними виконавчими елементами на виході застосовують транзисторний ключ (рис. 7.7, е), в якому резистор R є навантажувальним для елемента $DD1$ з вільним колектором і одночасно відіграє роль базового резистора $R_б$ ключа, а опір додаткового резистора R_1 вибирають з розрахунку, аби у вимкненому стані ключа струм через виконавчий елемент EL був недостатній для його вмикання.

Виходи кількох елементів з вільним колектором можна об'єднати на спільному навантаженні R (рис. 7.7, є) для реалізації так званої *монтажної логіки*. Рівень $y=1$ встановлюється лише при закритих транзисторах $VT4$ обох елементів, що відносно виходу еквівалентно функції монтажне І. Отже, відносно входів схема виконує функцію $y = x_1 x_2 x_3 x_4 = x_1 x_2 + x_3 x_4$, тобто двоступенева функція І-АБО-НІ реалізується без додаткових логічних елементів. Недолік монтажно-логіки – погіршення швидкодії внаслідок зростання паразитної навантажувальної ємності на спільному виході та повільний її заряд через зовнішній опір R .

3. Елементи з трьома станами виходу. Такий елемент (рис. 7.8, а) містить додатковий діод $VD1$ та керувальний вхід OE (Output Enable – дозвіл виходу). Якщо $OE=1$, то з'єднаний з цим входом емітерний перехід БЕТ $VT1$ та діод $VD1$ зачиняються, що еквівалентно відсутності їх у схемі, тому в цьому випадку елемент функціонує як звичайний, з двома станами на виході: $y=0$ або 1 залежно від сигналів на інформаційних входах x_1, x_2 (рис. 7.8, б). Якщо $OE=0$, зазначені перехід БЕТ та діод $VD1$ відкриваються, через БЕТ з елемента витікає вхідний струм $I_{вх}^0$, а

транзистори $VT2$, $VT4$ закриваються.

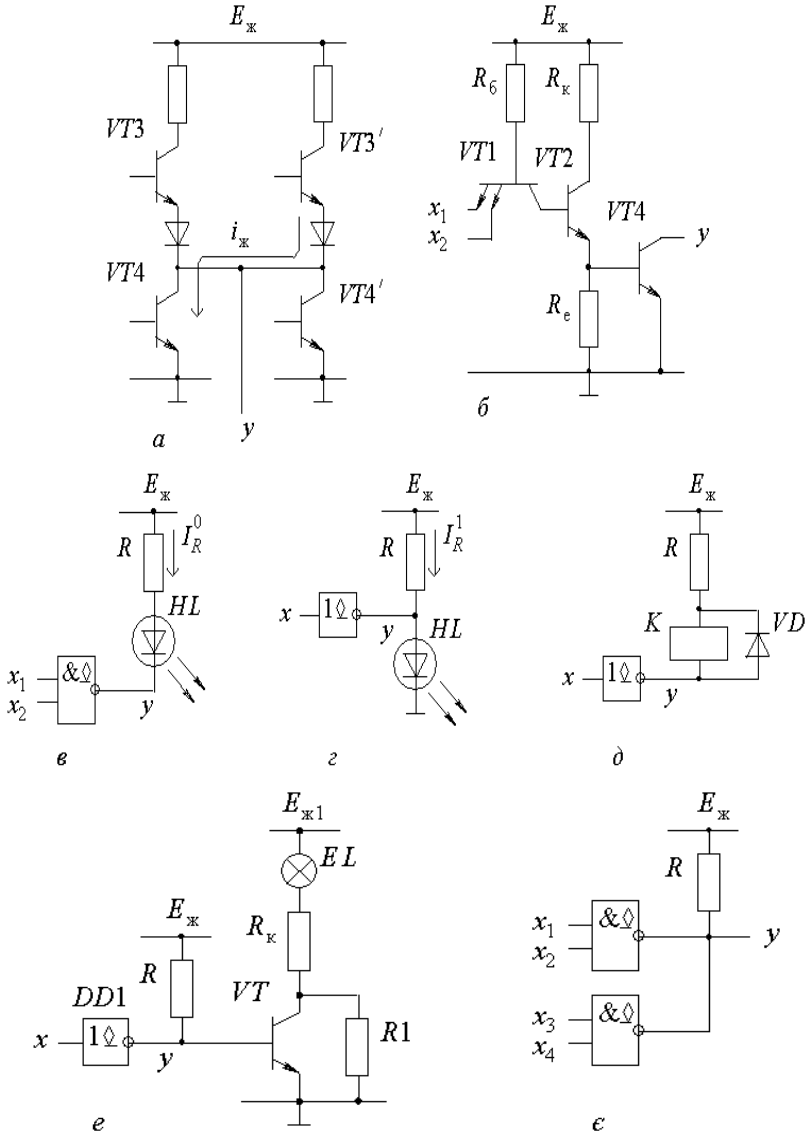


Рис. 7.7. Модифікації базового елемента ТТЛ

Однак на колекторі транзистора $VT2$ тепер не може вста-

новитися напруга високого рівня, тому що її величина $u_{K2}=U^0+U_{\text{пр}}<2U_{60}$ фіксується відкритим діодом $VD1$, внаслідок чого транзистор $VT3$ та діод VD також закриваються. З огляду на те, що обидва транзистори вихідного каскаду закриті, вихід y та з'єднана з ним лінія незалежно від логічних рівнів на входах x_1, x_2 опиняються у вільному стані, аналогічному розриву кола.

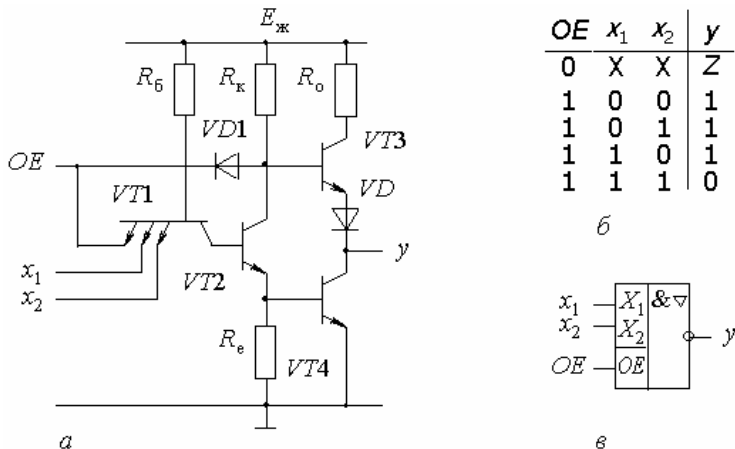


Рис. 7.8. Базовий елемент ТТЛ з трьома станами на виході

Такий третій стан БЕ називають також станом з великим опором або нескінченно великим імпедансом на виході. У таблиці відповідності його зазвичай позначають Z . Умовне графічне позначення елемента з входом дозволу OE подано на рис. 7.8, в. Елементи з трьома станами можна об'єднувати виходами до спільної лінії в режимі часового мультиплексування: керувальний сигнал $OE=1$, що дозволяє передавання даних лінією, у певний момент часу подається лише на один елемент – передавач інформації. При цьому на приймання інформації можуть працювати одночасно кілька елементів-приймачів.

Слід зауважити, що передавачі можна по черзі приєднувати до спільної лінії також за допомогою елементів з вільним колектором (див. рис. 7.7, е). При цьому в якості входу дозволу OE може використовуватися один із звичайних входів елемента. Під час дії на такому вході рівня лог. 0 (наприклад, $OE=x_2=0$ на

рис. 7.7, б) транзистор VT4 закривається і від'єнує вихід у від лінії, а рівнем лог. 1 ($OE=x_2=1$) сигнал з інформаційних входів надходить в лінію (у прикладі $y=x_1$).

4.Ементи з підвищеною навантажувальною здатністю. Коли до виходу елемента – джерела інформації підмикають лінію (магістраль) зі зв'язаними з нею входами багатьох БЕ-приймачів, зростають навантажувальна ємність C_H і вихідний струм БЕ-передавача, що може призвести до погіршення швидкодії та зміни вихідних рівнів у неприпустимих межах. З метою підвищення навантажувальної здатності у вихідному каскаді використовують *складений транзистор* VT3, VT6 (на рис. 7.9, а, всі транзистори та діоди можуть бути звичайними).

У базовій схемі (див. рис. 7.8, а), роль діода VD відіграє емітерний перехід транзистора VT6, підсилювальні властивості якого не використовуються. Шляхом об'єднання колекторів транзисторів VT3, VT6 дістаємо схему рис. 7.9, а, в якій роль діода зміщення VD відіграє той самий емітерний перехід транзистора VT6. Аби не зменшувати коефіцієнт підсилення транзистора VT3 та не збільшувати надмірно потужність транзистора VT6 (тому що його базовий струм має дорівнювати емітерному струму транзистора VT3), частину цього струму відгалужують через резистор R , що приводить до збільшення споживаної потужності елемента у вимкненому стані. Цей недолік усувається у варіанті схеми з підімкненим нижнім виводом резистора R до колектора транзистора VT4. Значний струм через резистор тепер протікає тільки під час перемикання елемента, а у вимкненому стані він через закритий транзистор VT4 не замикається, що зменшує споживану потужність.

Внаслідок збільшення підсумкового коефіцієнта підсилення за струмом емітерного повторювача на складеному транзисторі VT3, VT6 зменшується вихідний опір $R_{вих}^1$ елемента у вимкненому стані, тим самим зменшується стала часу заряду навантажувальної ємності C_H і підвищується навантажувальна здатність БЕ. Тому модифікації з повторювачем на складеному транзисторі називають елементами з підвищеним коефіцієнтом

розгалуження щодо виходу, з підвищеною навантажувальною здатністю, з потужним виходом. На умовних позначеннях елемент з підвищеною навантажувальною здатністю відрізняють додаванням трикутної позначки підсилення (рис. 7.9, б).

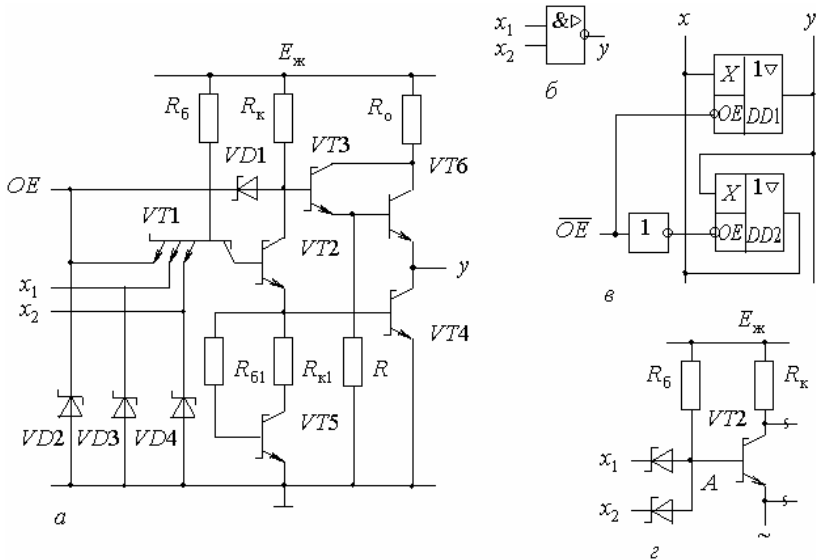


Рис. 7.9. Базовий елемент ТТЛ з підвищеною навантажувальною здатністю

5. Магістральні елементи. БЕ з потужним виходом та з трьома станами виходу (див. рис. 7.9, а) називають також магістральними елементами або драйверами. При цьому вхід дозволу OE може бути інверсним, якщо керувальний сигнал надходить через вбудований в елементі інвертор. Об'єднанням двох магістральних елементів за схемою рис. 7.9, в можна дістати двоспрямований елемент (приймач-передавач): при $\overline{OE}=0$ сигнал з лінії x надходить до лінії y через елемент $DD1$, а при $\overline{OE}=1$ – у зворотному напрямку через елемент $DD2$.

6. Елементи ТТЛШ. Існують малопотужні серії ТТЛ, в яких для зменшення споживаного від джерела живлення струму застосовують резистори з більшим опором і, навпаки, в швидкодійних серіях з метою збільшення перемикальних струмів та

зменшення сталих часу паразитних ємностей використовують резистори з малим опором. При цьому енергія перемикавання $A = P_{\text{жс}} \bar{t}_{\text{з.н}}$ (добуток споживаної потужності на середній час затримки поширення) залишається приблизно постійною, тому цей показник характеризує рівень технології.

Якісний стрибок зроблено шляхом переходу до технології ТТЛ на транзисторах і діодах Шотткі (ТТЛШ). Елементи сучасних серій ТТЛШ відрізняються більшою швидкодією при меншій споживаній потужності і мають приблизно на порядок меншу енергію перемикавання. Впроваджено елементи ТТЛШ (див. рис. 7.9, *а*) зі всіма схемними модифікаціями: звичайний БЕ, БЕ з трьома станами виходу – з діодом $VD1$ і керувальним входом OE , з демпферними діодами $VD2$, $VD3$, $VD4$, з коригувальним каскадом $VT5$, з потужним виходом – зі складеним транзистором $VT3$, $VT6$ (транзистор $VT6$ виконують звичайним, тому що він не входить до режиму насичення), а також з вільним колектором.

7. Малопотужні швидкодіїні елементи ТТЛШ. У малопотужних серіях ТТЛШ в схемі рис. 7.9, *а* замість БЕТ використовують діодну збірку (рис. 7.9, *з*), аналогічну БЕ ДТЛ (див. рис. 7.1, *а*), але без діодів зміщення, роль яких відіграє емітерний перехід транзистора $VT2$. Незважаючи на те, що в колі зміщення на один перехід менше (відсутній перехід $V_{\text{БК}}$ транзистора $VT1$), завадостійкість майже не погіршується, тому що напруження в точці A при вхідному рівні лог. 0 на діоді Шотткі нижча. Проте при цьому покращується розв'язка між елементами: під час дії вхідного рівня лог. 1 діоди відмикають входи БЕ від джерела сигналу. З огляду на те, що технологія і стандартні рівні елементів зберігаються, їх прийнято відносити до типу ТТЛШ.

У модифікованих малопотужних швидкодіїних серіях ТТЛШ діодну збірку підімкнено до фазорозділювального каскаду $VT2$ через емітерний повторювач. Крім того, за допомогою додаткових діодів Шотткі, ввімкнених у вхідному і вихідному каскадах, досягається форсування процесів перемикавання БЕ.

7.1.6. Розрахунок сполучення кіл з елементами ТТЛ

1. Розрахунок струмового навантаження. При наван-

таженні БЕ ТТЛ(Ш) на магістраль або на розгалужене коло з'єднань з різними елементами, наприклад, на коло скидання тригерів, лічильників тощо необхідно перевірити, чи задовольняє вимоги навантажувальна здатність БЕ. Для цього розраховують реальне струмове навантаження елементу ($DD1$ на рис. 7.10, *a*), виходячи з того, що при об'єднанні входів БЕ ТТЛ вхідний струм не змінюється при $x=0$, а при $x=1$ вхідні струми підсумовуються:

$$I_{ex\Sigma}^0 = I_{ex}^0; \quad I_{ex\Sigma}^1 = K_{об} I_{ex}^1, \quad (7.4)$$

де $K_{об}$ – коефіцієнт об'єднання входів. Тому вихідні струми елементу $DD1$, навантаженого на N аналогічних елементів, розраховують за такими співвідношеннями:

$$I_{eu\Sigma}^0 = N I_{ex}^0; \quad I_{eu\Sigma}^1 = I_{ex}^1 \sum_{i=1}^N K_{об.i}, \quad (7.5)$$

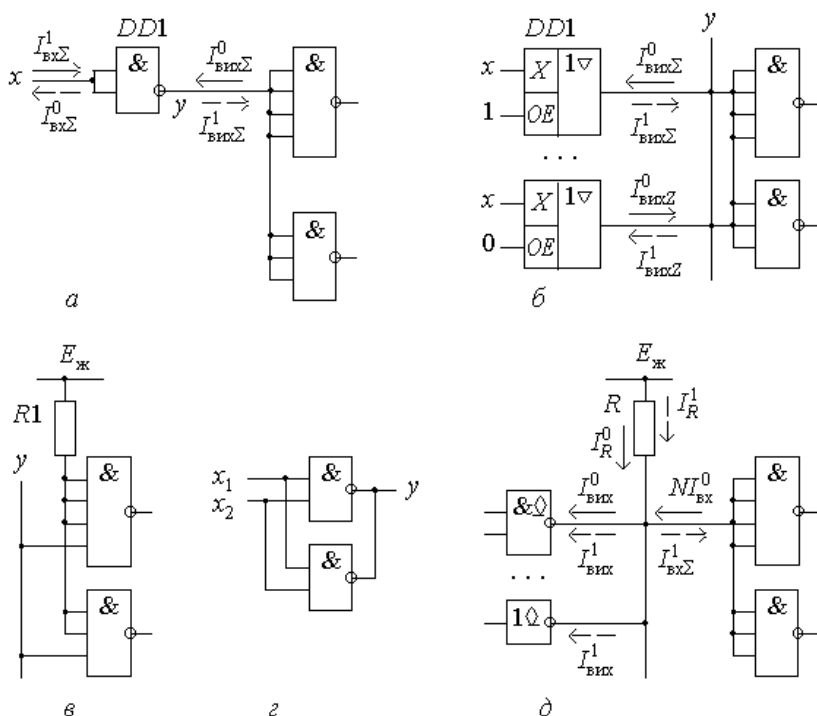


Рис. 7.10. До розрахунку струмового навантаження БЕ ТТЛ

Приклад 7.1. Для схеми рис. 7.10, а при $I_{\text{ex}}^0 = 1 \text{ мА}$, $I_{\text{ex}}^1 = 40 \text{ мкА}$, $N=2$ за (7.4) для елемента DD1 маємо $I_{\text{ex}\Sigma}^0 = I_{\text{ex}}^0 = 1 \text{ мА}$, $I_{\text{ex}\Sigma}^1 = 2I_{\text{ex}}^1 = 0.08 \text{ мА}$ та згідно з (7.5) $I_{\text{вix}\Sigma}^0 = 2I_{\text{ex}}^0 = 2 \text{ мА}$, $I_{\text{вix}\Sigma}^1 = I_{\text{ex}}^1 \cdot (4 + 3) = 0.28 \text{ мА}$.

При об'єднанні виходів M елементів з *трьома станами* на спільну магістраль у (рис. 7.10, б) додатково враховують їх струми витоку. Коли сигналом $OE=1$ активовано лише елемент DD1, а інші $M-1$ елементів-передавачів під дією рівня $OE=0$ перебувають у третьому стані, то при $y=0$ струми втрати $I_{\text{вix}Z}^0$ витікають з елементів у третьому стані, а при $y=1$ струми $I_{\text{вix}Z}^1$ втікають до них. Отже, відносно виходу елементу DD1 з урахуванням (7.5) маємо таке струмове навантаження:

$$I_{\text{вix}\Sigma}^0 = NI_{\text{ex}}^0 + (M - 1)I_{\text{вix}Z}^0; \quad (7.6)$$

$$I_{\text{вix}\Sigma}^1 = I_{\text{ex}}^1 \sum_{i=1}^N K_{\text{об.і}} + (M - 1)I_{\text{вix}Z}^1. \quad (7.7)$$

Приклад 7.2. Для схеми рис. 7.10, б при навантаженні як у попередньому прикладі та $M=11$, $I_{\text{вix}Z}^0 = I_{\text{вix}Z}^1 = 20 \text{ мкА}$ розраховуємо навантаження елементу DD1, на вхід дозволу якого подано рівень $OE=1$. За (7.6) та (7.7): $I_{\text{вix}\Sigma}^0 = 2I_{\text{ex}}^0 + 10I_{\text{вix}Z}^0 = 2.2 \text{ мА}$; $I_{\text{вix}\Sigma}^1 = I_{\text{ex}}^1 \cdot (4 + 3) + 10I_{\text{вix}Z}^1 = 0.48 \text{ мА}$.

У всіх випадках струми $I_{\text{вix}\Sigma}^0$ та $I_{\text{вix}\Sigma}^1$ мають не перевищувати паспортних гранично допустимих величин, причому при об'єднанні виходів різних ІС розрахунки за (7.6) та (7.7) виконують для найменш потужної з них.

Струмове навантаження можна зменшити підімкненням зайвих входів БЕ до джерела живлення через резистор опором $R_I = 1 \text{ кОм}$, причому до одного резистора можна підімкнути до 20 входів БЕ ТТЛ. З'єднання навантажувальних елементів на рис. 7.10, б за таким способом для прикладу наведено на рис.

7.10, в. Константу лог. 1 зручно утворювати і за допомогою додаткового елементу, наприклад, І-НІ (зокрема, незадіяного в корпусі ІС) з'єднанням його входів з спільним проводом "землею". Якщо це потужний елемент, до його виходу можна приєднати до 30 входів БЕ ТТЛ(Ш). Якщо ж у підсумку фактичне струмове навантаження виявляється більшим за паспортне, застосовують елементи з підвищеною навантажувальною здатністю. Допускається також об'єднувати виходи кількох БЕ за умови об'єднання їх входів (рис. 7.10, з); при об'єднанні двох елементів навантажувальна здатність збільшується в 1,9 рази: $K_{роз} = 1.9 K_{роз}$.

2. Розрахунок зовнішнього опору елемента з вільним колектором. Опір резистора R у колі елемента з вільним колектором (див. рис. 7.7, в...є) розраховують, виходячи з струму світіння світлодіода, спрацювання реле, насичення транзистора VT тощо таким чином, аби струм $I_{вих}^0$ БЕ не перевищував граничного значення. При цьому для схеми рис. 7.7, в при вмиканні виконавчого елемента рівнем лог. 0 на виході отримуємо:

$$R = \frac{E_{жс} - U_{np} - U^0}{I_R^0}, \quad (7.8)$$

де U_{np} – напруга прямого зміщення світлодіода при $y=0$; I_R^0 – прямий струм світлодіода, що забезпечує потрібну яскравість світіння і вибирається з характеристики яскравості за умов $I_R^0 < I_{вихдон}^0$, $I_R^0 < I_{нр.дон}$ ($I_{вихдон}^0$, $I_{нр.дон}$ – гранично допустимі струми БЕ та світлодіода).

Аналогічно для схеми рис. 7.7, з при вмиканні виконавчого елемента рівнем лог. 1 на виході отримаємо:

$$R = \frac{E_{жс} - U_{np}}{I_R^1}, \quad (7.9)$$

де $I_R^1 = I_{np}^1 + I_{вих}^1$. Крім того, коли транзистор $VT4$ відкривається (світлодіод гасне), має виконуватися умова

$$R \geq \frac{E_{жс} - U^0}{I_{вихдон}^0}, \quad (4.10)$$

Приклад 7.3. Розрахуємо кола індикації на БЕ типу К155ЛА7 та світлодіоді АЛ301Б з такими параметрами: $I_{вих}^0 \leq 30 \text{ мА}$, $I_{вих}^1 \leq 0.25 \text{ мА}$, $I_{np} \leq 11 \text{ мА}$. Припускаючи $U^0 = 0.3 \text{ В}$ та за характеристикою світіння яскравість $B=20 \text{ нт}$ при $I_{np}=6 \text{ мА}$, $U_{np}=3,3 \text{ В}$, для схеми рис. 7.7, в при $I_R^0 = I_{np}$ згідно з (7.8) маємо $R=233 \text{ Ом}$; вибираємо $R=220 \text{ Ом}$. Для схеми рис. 7.7, з: $I_R^1 = 6.25 \text{ мА}$, за (7.9) $R=272 \text{ Ом}$; вибираємо $R=300 \text{ Ом}$, тому що при цьому виконується умова (7.10): $R>153 \text{ Ом}$.

Розрахунок навантажувального резистора R при об'єднанні виходів елементів з вільним колектором (рис. 7.10, д) виконують таким чином. Опір цього резистора має обмежувати вихідний струм $I_{вих}^0 = I_R^0 + NI_{ех}^0$, звідки

$$I_R^0 = I_{вих}^0 - NI_{ех}^0; \quad (7.11)$$

при цьому вихідна напруга має не перевищувати максимальний рівень лог. 0:

$$U^0 = E_{ж.макс} - I_R^0 R \leq U_{макс}^0 \quad (7.12)$$

При вимкнених усіх M елементах з вільним колектором з об'єднаними виходами струм через резистор R замикається до їх виходів та входів БЕ-навантажень:

$$I_R^1 = MI_{вих}^1 + I_{ех}^1 \sum_{i=1}^N K_{об.i} \quad (7.13)$$

і вихідна напруга має бути не нижчою мінімально допустимого рівня лог. 1:

$$U^1 = E_{ж.мін} - I_R^1 R \geq U_{мін}^1 \quad (7.14)$$

Згідно з (7.12) та (7.13) опір R розраховуємо за виразом

$$\frac{E_{ж.макс} - U_{макс}^0}{I_R^0} \leq R \leq \frac{E_{ж.мін} - U_{мін}^1}{I_R^1}. \quad (7.15)$$

Приклад 7.4. Розрахуємо опір R для схеми, наведеної на рис. 7.10, д при $M=2$, параметрах БЕ з вільним колектором: $I_{вих}^0 = 16 \text{ мА}$, $I_{вих}^1 = 0.25 \text{ мА}$, навантаженням як у прикладах п.1

та нестабільністю напруги джерела живлення $\pm 5\%$.

За (7.11), (7.13) маємо $I_R^0 = 12.8 \text{ мА}$, $I_R^1 = 0.78 \text{ мА}$; згідно з (7.15) $0,38 \text{ кОм} \leq R \leq 3,01 \text{ кОм}$. З міркувань швидкодії вибираємо опір ближчим до меншої межі: $R = 470 \text{ Ом}$.

Таким чином, базовою є схема елементу ТТЛ зі складним інвертором, яка виконує функцію І-НІ. Середній час затримки поширення під час перемикавання універсального елементу ТТЛ становить порядку 10 нс ; в обох станах забезпечується низький вихідний опір, завдяки чому елемент здатний працювати на досить значну ємність навантаження. Внаслідок виникнення сплеску струму в шині живлення протягом перемикавання утворюються завади, тому для їх придушення вдаються до конструктивних заходів (раціонально розташовують з'єднувальні шини, застосовують фільтрувальні конденсатори в колі живлення).

У цифровій та мікропроцесорній техніці поширені різні модифікації елементу, зокрема, з трьома станами виходу, що дозволяє об'єднувати кілька елементів – передавачів інформації до спільної лінії в режимі часового мультиплексування. З метою підвищення швидкодії без збільшення споживаної потужності впроваджено серії ТТЛШ, в яких енергія перемикавання менша, ніж в елементів ТТЛ (середні значення параметрів деяких серій подано в табл. 7.1). Елементи ДТЛ, що застосовуються в сучасних малопотужних, а також завадостійких серіях, зазвичай відносять до класу ТТЛ, тому що за параметрами і технологією вони сумісні з елементами ТТЛ.

Таблиця 7.1. Порівняльна характеристика
БЕ ТТЛ різних серій

Серія	Призначення серії	$t_{з.п}, \text{ нс}$	$P_{ж}, \text{ мВт}$	$A, \text{ нДж}$
155	Універсальна, ТТЛ	10	10	100
531	Швидкодійна, ТТЛШ	3	20	60
533, 555	Малопотужна, ТТЛШ	10	2	20
1531	Швидкодійна малопотужна, ТТЛШ	3	4	12
1533	Швидкодійна малопотужна, ТТЛШ	4	2	8

7.2. Базові елементи ЕСЛ

7.2.1. Перемикач струму

Головна перевага ІС емітерно-сполученої логіки (ЕСЛ) полягає в їх великій швидкодії. Завдяки функційній та схемотехнічній гнучкості, відпрацьованості технології ІС ЕСЛ набули поширення як елементна база швидкодійних пристроїв; засвоєно випуск біля 50 типоміналів ІС.

Основою БЕ ЕСЛ є підсилювальний диференційний каскад (ДК) за схемою з несиметричним входом e та несиметричними виходами $u_{к1}$, $u_{к2}$ (рис. 7.11, а). До складу ДК входять керувальне плече на транзисторі $VT1$ і опорне – на $VT2$, до входу якого прикладено постійну напругу E_0 від джерела опорної напруги (ДОН), що живиться від джерела живлення $E_{ж}$. Завдяки ДОН і негативному зворотному зв'язку за струмом, який здійснюється через опір R_e , забезпечується стабілізація параметрів ДК під час зміни температури середовища або напруги живлення, а також при старінні елементів.

Як видно зі схеми, в ДУ завжди виконується співвідношення

$$u_e = e - u_{\delta e1} = E_0 - u_{\delta e2}. \quad (7.16)$$

З огляду на те, що на емітерному переході відкритого транзистора напруга U_{np} майже незмінна, емітерний струм можна обчислити як

$$I_e = \frac{u_e}{R_e} = \frac{E_0 - U_{np}}{R_e}, \text{ тобто він визначається}$$

приблизно сталими величинами.

Тому емітерне коло на еквівалентній схемі (рис. 7.11, б) представлене генератором стабільного струму (ГСС) I_e . Практично ГСС реалізується у вигляді опору R_e (у серії 500) або транзисторного каскаду, що живиться від ДОН (у серії 1500).

За однакових напруг $e=E_0$ на входах ДК обидва транзистори відкриваються і при симетричній схемі через них протікають струми $i_{к1}=i_{к2}=h_{21\delta}I_e/2 \approx I_e/2$ та на виходах встановлюються

напруги $u_{к1}=u_{к2}\approx E_{ж} - e R_{к}/2$. Цей, збалансований щодо виходів еквівалентного мосту стан, відповідає зображувальній точці 1 на передатних характеристиках ДК (рис. 7.11, в). Під час зменшення напруги e відносно E_0 на деяку малу величину Δu опорне плече залишається відкритим і напруги $u_{бe2}=U_{пр}$ та $u_e=E_0-U_{пр}$ майже незмінними.

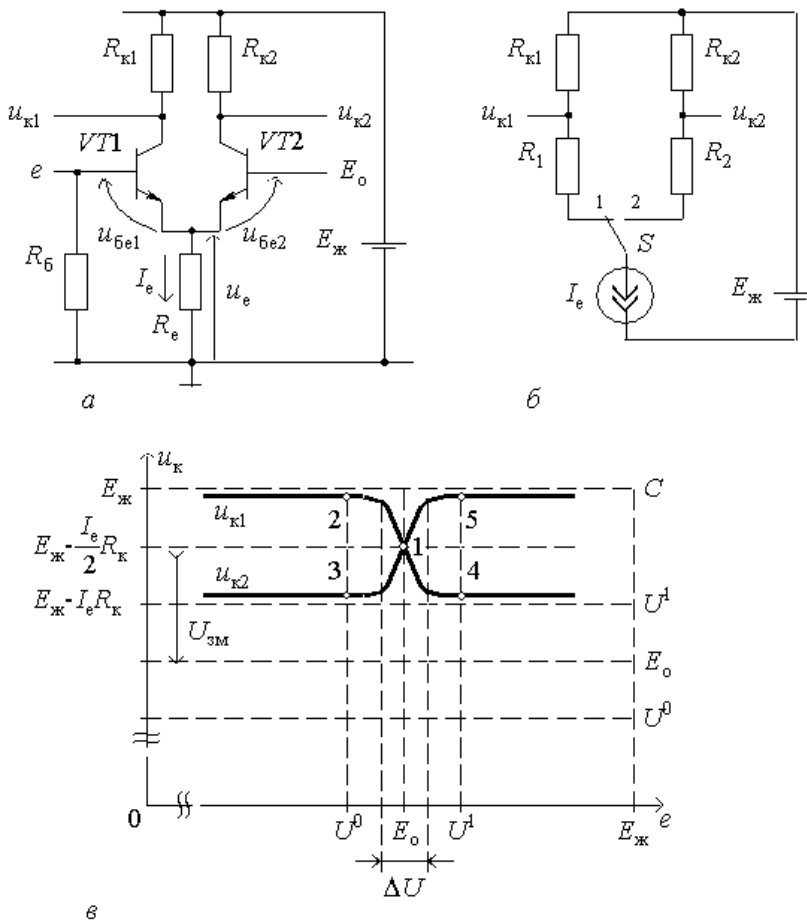


Рис. 7.11. Перемикач струму на диференціальному каскаді

Тому згідно з (7.16) величина $u_{бe1}=e-U_e$ зменшується,

транзистор $VT1$ починає закриватися, практично сталий струм $I_e = i_{e1} + i_{e2}$ перерозподіляється: i_{e1} зменшується, а i_{e2} збільшується, наслідок чого зниження рівня e спричиняє збільшення напруги $u_{k1} = E_{ж} - i_{e1}R_{k1}$ та зменшення $u_{k2} = E_{ж} - i_{e2}R_{k2}$. Отже, вихід керувального плеча є *інверсним*, а опорного – *прямим*.

Якщо негативний приріст вхідної напруги сягає величини розмаху вхідної характеристики $\Delta U = U_{пр} - U_{б0} \approx (0,1 \dots 0,2) B$, транзистор $VT1$ закривається і струм I_e повністю перемикається до транзистора $VT2$, що відповідає положенню 2 перемикача S на спрощеній еквівалентній схемі (див. рис. 7.11, б), на якій внутрішні опори транзисторів умовно позначено R_1 та R_2 . Колекторні напруги u_{k1} , u_{k2} при цьому набувають значення

$$U_{к1}^1 \approx E_{ж}; U_{к2}^0 \approx E_{ж} - I_e R_{к2}, \quad (7.17)$$

якщо знехтувати спадом напруг на резисторах $R_{к1}$, $R_{к2}$, спричинених відгалуженням незначних струмів у навантаження. Такий стан ДК відображається точками 2, 3 на передатній характеристиці.

І, навпаки, під час перевищення вхідною напругою величини E_0 транзистор $VT1$ відкривається, напруга на його емітерному переході $u_{бe1} = U_{пр}$ лишається практично сталою, тому весь позитивний приріст Δe передається на емітер, і згідно з (7.16) напруга $u_{бe2} = E_0 - u_e$ зменшується, тобто транзистор $VT2$ починає закриватися, а відтак, при досягненні позитивного приросту $\Delta e \geq \Delta U$ весь струм I_e перерозподіляється до транзистора $VT1$, тобто перемикач S на рис. 7.11, б опиняється в положенні 1. Колекторні напруги відповідно до (7.17) набувають протилежного значення (точки 4, 5 на рис. 7.11, в).

Таким чином, ДК підсилює диференційну напругу $u_d = e - E_0$ – різницеву між його входами. Якщо керувальна напруга e змінюється в межах не менше, ніж розмах вхідної характеристики транзистора DU , ДК працює в ключовому режимі і є *перемикачем струму* (ПС), тому що емітерний струм I_e під дією сигналу e по черзі перемикається до транзисторів $VT1$ та $VT2$. З огля-

ду на незначну величину навколопорогової зони DU пороговою напругою перемикавання можна вважати значення E_0 , вхідним рівнем лог. 1 – напругу $e=U^1>E_0+\Delta U$, що перевищує опорну напругу E_0 на величину, не нижчу, ніж ΔU , а вхідним рівнем лог. 0 – напругу $e=U^0<E_0-\Delta U$, меншу за E_0 на таку саму величину.

З метою підвищення швидкодії в ПС вибирається *активний* режим відкритих транзисторів. Для цього напруга E_0 і вхідні логічні рівні U^0 , U^1 (див. рис. 7.11, в) мають бути нижчими, ніж колекторні напруги $u_{к1}$ та $u_{к2}$, тоді колекторні переходи транзисторів ПС завжди зміщені у зворотному напрямку.

7.2.2. Основна схема

Керувальне плече ПС у складі БЕ ЕСЛ (рис. 7.12, а) утворюється транзисторами $VT1.1$, $VT1.2$ – за кількістю входів x_1 , x_2 для реалізації логічної функції АБО, а опорне плече – транзистором $VT2$. Резистори R_1 , R_2 призначені для певного закривання незадіяних вхідних транзисторів, а також для стікання зворотного базового струму під час їх закривання, через що зайві входи БЕ ЕСЛ можна залишати вільними. З колекторних навантажень ПС сигнали надходять до інверсного y_1 та прямого y_2 виходів (рис. 7.12, б) через емітерні повторювачі $VT3$ та $VT4$, які для розширення функціональних можливостей виконують без навантажень R_3 , R_4 всередині БЕ. Резистори можна підключити ззовні елементу, як зображено пунктиром, або іншим способом.

Якщо колекторні виходи ПС підключити безпосередньо до входів керованих БЕ-навантажень, то транзистори $VT1$ елементу-навантаження завжди будуть відкритими, а $VT2$ – закритими, тому що напруги $u_{к1}$, $u_{к2} > E_0$ (див. рис. 7.11, в). Емітерні переходи транзисторів $VT3$, $VT4$ виконують функцію *зміщення* колекторних напруг на величину $U_{зм}=U_{пр}$ таким чином, що вхідні напруги u_{y1} , u_{y2} змінюються відносно опорної E_0 , тобто взмозі перемикати БЕ-навантаження. Крім узгодження напруг емітерні повторювачі підвищують навантажувальну здатність

БЕ. Транзистори $VT3$, $VT4$ також перебувають в активному режимі: їх базові напруги не можуть перевищувати напруги на колекторах, підключених до позитивного полюсу джерела живлення.

У БЕ ЕСЛ, як і в розглянутому ПС, джерело $E_{\text{ж}}$ підключено так само: верхня за схемою шина з'єднана з позитивним полюсом, а нижня – з негативним, тому струми і спади напруг на ділянках кіл лишаються такими самими. Різниця полягає в тому, що в БЕ ЕСЛ заземлено верхню шину, тобто використовується джерело живлення негативної напруги $-E_-$. Завдяки цьому зменшується вплив нестабільності джерела живлення і завад, що поширюються шиною живлення, адже тепер при закритому одному з плечей ПС напруга на його виході дорівнює не $E_{\text{ж}}$, а нулю; на виході протилежного, відкритого плеча напруга значно менша, ніж при живленні БЕ від джерела $+E_{\text{ж}}$. Крім того, під час випадкового закорочування на корпус виходу схеми струм протікає через зовнішні резистори R_3 , R_4 , а не через транзистори $VT3$, $VT4$ і, отже, немає потреби захищати їх від виходу з ладу за допомогою обмежувальних резисторів в колекторних колах, як це робиться, наприклад, у БЕ ТТЛ.

7.2.3. Статичні режими та передатна характеристика

Передатну характеристику ПС при негативній напрузі живлення отримаємо, якщо точку перетину C ліній, що виходять з $+E_{\text{ж}}$ (див. рис. 7.11, *в*) приймемо за нуль (рис. 7.12, *в*).

Дійсно, у *вимкненому* відносно керувального плеча стані БЕ, тобто при $x_1 = x_2 = 0$, до всіх входів надходять напруги низького рівня $U^0 < U_{\text{п}} = E_0$, тому всі транзистори керувального плеча $VT1.1$, $VT1.2$ закриті, а транзистор $VT2$ опорного плеча відкритий і перебуває в активному режимі. Отже, весь струм $I_{\text{е}}$ при цьому протікає через опорне плече: $i_{\text{к1}} \approx 0$, $i_{\text{к2}} \approx I_{\text{е}}$, тому напруги $u_{\text{к1}}$, $u_{\text{к2}}$ на виходах ПС при цьому дорівнюють відповідно

$$U_{\text{к}}^1 \approx 0; \quad U_{\text{к}}^0 \approx -I_{\text{е}} R_{\text{к}}, \quad (7.18)$$

що відображається на передатній характеристиці ПС (див. рис.

7.12, в) точками 2, 3.

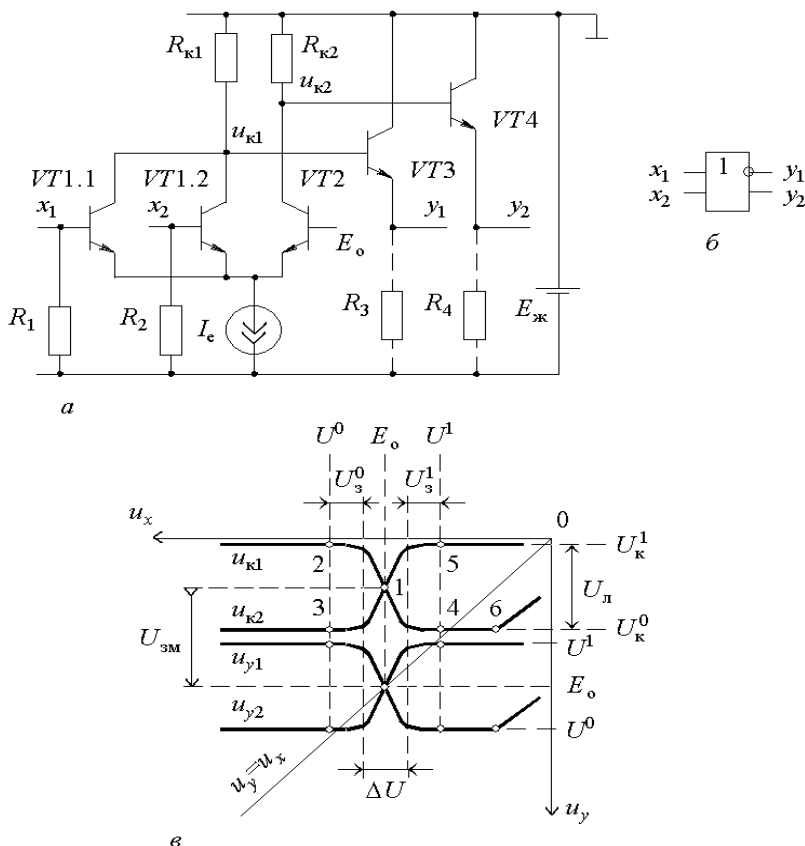


Рис. 7.12. Базовий елемент ЕСЛ

У ввімкненому стані, коли хоча б на один зі входів $x_i=1$ надходить напруга високого рівня $U^1 > U_{\pi} = E_0$, відкривається і переходить до активного режиму хоча б один з транзисторів VT1.1, VT1.2 керувального плеча, а транзистор VT2 закривається, струм I_e перемикається до резистора R_{K1} , а через R_{K2} не протікає: $i_{K1} \approx I_e$, $i_{K2} \approx 0$, отже, напруги u_{K1}, u_{K2} змінюються відносно (7.18) на протилежні значення, що відповідає зображувальним точкам 4, 5.

Під час зменшення напруг на входах x_1, x_2 відносно U^0 транзистори VT1.1, VT1.2 лишаються закритими, тому напруги $u_{к1}, u_{к2}$ ліворуч від точок 2, 3 не змінюються. У навколопороговій області завширшки DU , що дорівнює розмаху вхідної характеристики, транзистори обох плечей потрапляють до активного режиму і відбувається перемикання ПС: на характеристиках точка 2 прямує до 4, а точка 3 – до 5. Під час подальшого збільшення вхідної напруги хоча б на одному з входів x_i відповідний транзистор VT1.1 або VT1.2 відкривається все більше і, коли (в точці 6) напруга на його базі перевищує колекторну $u_{к1}$, цей транзистор опиняється в режимі насичення; його колекторний перехід відкривається і подальший приріст вхідної напруги надходить до виходу $u_{к1}$. Цим зумовлений злам характеристики в точці 6, праворуч від якої ділянка є неробочою.

З огляду на те, що емітери повторювачів VT3, VT4 підімкнені до негативного полюсу джерела живлення, їх базові напруги $u_{б3}=u_{к1}, u_{б4}=u_{к2}$ завжди перевищують емітерні, тому ці транзистори завжди відкриті і перебувають в активному режимі. Отже, вихідні напруги u_{y1}, u_{y2} повторюють значення $u_{к1}, u_{к2}$ зі зміщенням їх на величину спаду напруги на емітерних переходах $U_{зм}=U_{пр}$:

$$U^1=U^I_{к}-U_{пр}\approx -U_{пр}; U^0=U^0_{к}-U_{пр}\approx -I_{е}R_{к}-U_{пр}.$$

Перетин передатних характеристик з лінією одиничного підсилення $u_y=u_x$ в точці, що відповідає вхідним і вихідним напругам E_0 , свідчить про узгодженість рівнів. Через те, що логічний перепад $U_{л}=U^1-U^0\approx U_{пр}$ невеликий, завадостійкість слід визначати за верхнім і нижнім пороговими рівнями $U_{п}=E_0\pm\pm DU/2$, тобто відносно точок зламу характеристик завадостійкість становить $U^0_3=U^I_3\approx 0,5(U_{л}-\Delta U)$. З урахуванням розкиду характеристик та нестабільності напруги живлення реальна завадостійкість в діапазоні температур становить (125...150) мВ. Відносна завадостійкість $U_3/U_{л}\approx(15...20)\%$ наближається до БЕ

ТТЛ і достатня завдяки стабілізації електричного режиму й тому, що через ПС протікає майже сталий струм I_e і в ньому не утворюються завади протягом перемикування. Імпульсні завади можуть виникати в емітерних повторювачах; для зменшення їх впливу застосовують окреме підімкнення спільної шини повторювачів і ПС.

7.2.4. Особливості сполучення із зовнішніми колами

З метою розширення функціональних можливостей на виходах БЕ ЕСЛ вмикають кілька емітерних повторювачів паралельним з'єднанням баз та колекторів вихідних транзисторів. Таким чином утворюється БЕ з багатоемітерними транзисторами; фрагмент схеми з БЕТ керувального плеча ПС наведено на рис. 7.13, а. Шляхом з'єднання емітерів БЕТ *одного* БЕ отримують *елемент з потужним виходом*, а емітерним об'єднанням *різних* БЕ реалізують операцію *монтажне АБО*. Так, при об'єднанні прямих виходів y_1, y_2 двох БЕ (рис. 7.13, б) до зовнішнього навантаження R_1 на спільному виході z_1 встановлюється високий рівень, якщо високий потенціал надходить хоча б з одного виходу: $z_1 = y_1 + y_2 = x_1 + x_2 + x_3 + x_4$, тобто емітерне об'єднання прямих виходів еквівалентне збільшенню коефіцієнта об'єднання $K_{об}$. Об'єднанням інверсних виходів $\overline{y_1}, \overline{y_2}$ елементів до навантаження R_2 реалізується *функція АБО-І-НІ*: $z_2 = \overline{y_1 + y_2} = \overline{y_1 y_2} = \overline{(x_1 + x_2) \cdot (x_3 + x_4)}$. Подібні емітерні об'єднання можуть виконуватися і всередині ІС.

При проектуванні цифрових пристроїв слід враховувати, що на виході БЕ ЕСЛ формуються імпульси з фронтами наносекундної тривалості, тобто в спектрі сигналів містяться коливання НВЧ з короткою довжиною хвилі. Тому навантаження, віддалене від виходу БЕ на відстань, сумірну з довжиною хвилі, з'єднують за допомогою ліній зв'язку з хвильовим опором, що становить, здебільшого, величину $r=50, 75$ або 100 Ом . Для зменшення відбиття при поширенні сигналів лінією зв'язку її *узгоджують із навантаженням*, у найпростішому випадку за допомогою навантажувального для емітерного повторювача VT3 ре-

зистора $R_3=r$ (рис. 7.13, а), який приєднують до джерела живлення $E_{ж1}$ у кінці лінії, безпосередньо біля входу навантажувального БЕ.

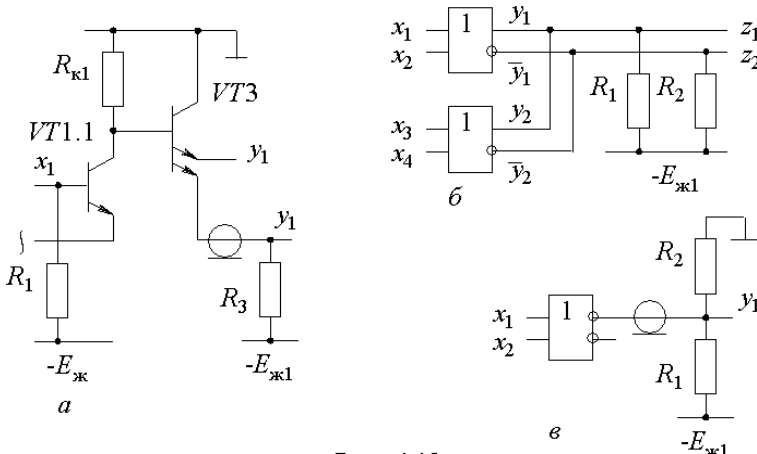


Рис. 7.13. Реалізація базовим елементом ЕСЛ різних функцій

Використання низькоопірних навантажувальних резисторів одночасно сприяє підвищенню швидкодії, проте призводить до збільшення споживаної потужності внаслідок зростання струму через емітерний повторювач

$$I_{\text{вих}}^1 = (U^1 - E_{ж1}) / R_3, \quad (7.19)$$

який при застосуванні стандартного для БЕ ЕСЛ джерела живлення $E_{ж1} = E_{ж} = -5,2 \text{ В}$ та при рівні лог. 1 на виході може перевищити гранично допустимий рівень. З метою уникнення цього в колі емітерних повторювачів використовують окреме джерело живлення зі зменшеною напругою, наприклад, $E_{ж1} = (-2; -2,4) \text{ В}$.

Приклад 7.5. Для узгодження з лінією зв'язку, що має хвильовий опір $\rho = 50 \text{ Ом}$, за схемою рис. 7.13, а застосовуємо резистор $R_3 = \rho = 50 \text{ Ом}$. При використанні спільного джерела живлення $E_{ж1} = E_{ж} = -5,2 \text{ В}$ за (7.19) маємо $I_{\text{вих}}^1 = 86 \text{ мА}$. Для зменшення струму через емітерний повторювач вибираємо напругу для його живлення $E_{ж1} = -2 \text{ В}$ або за відсутністю такого джерела вико-

ристовуємо подільник напруги (рис. 7.13, в), опори якого мають задовольняти умови:

$$E_{ж1} = E_{ж2} R_2 / (R_1 + R_2); \quad R_1 R_2 / (R_1 + R_2) = \rho,$$

звідки дістаємо $R = \rho E_{ж1} / E_{ж2} = 130 \text{ Ом}$; $R_2 = \rho R_1 / (R_1 - \rho) = 81,25 \text{ Ом}$ та вибираємо $R_1 = 130 \text{ Ом}$, $R_2 = 82 \text{ Ом}$. Перевіряємо вихідний струм за (7.19) при $E_{ж1} = -2 \text{ В}$: $I_{вих}^1 = 22 \text{ мА}$, що не перевищує гранично допустимого значення для БЕ ЕСЛ.

Таким чином, основу БЕ ЕСЛ складає диференційний каскад, що працює в режимі перемикача струму. Вихід керувального плеча БЕ ЕСЛ є інверсним, а опорного – прямим і відносно парафазних виходів реалізується логічна функція АБО-НІ/АБО в позитивній логіці (вищий з урахуванням знаку рівень напруги кодується лог. 1, а нижчий – лог. 0).

Головна перевага БЕ ЕСЛ – висока швидкодія – зумовлена тим, що: 1) відкриті транзистори перебувають в активному режимі, тому відсутній час розсмоктування надлишкового заряду; 2) транзистори завдяки розвиненості технології виконуються з високою граничною частотою; 3) паразитні ємності перезаряджаються в низькоопірних колах: навантажувальні резистори вибираються з малими опорами і зв'язок між БЕ здійснюється через емітерні повторювачі з малим вихідним опором.

Проте зменшення номіналів резисторів приводить до збільшення споживаної потужності, більша частина якої витрачається емітерними повторювачами. Незважаючи на те, що за споживаною потужністю БЕ ЕСЛ перевищують інші типи ІС, енергія перемикання $A = P_{ж3,п} t_{3,п}$ у БЕ ЕСЛ менша, тобто для досягнення такої самої швидкодії інші ІС потребують не меншої споживаної потужності. Застосування БЕ ЕСЛ ефективно й доцільно не лише в швидкодіючих пристроях.

7.3. Базові елементи на МОН-структурах

7.3.1. Базовий елемент МОНТЛ

БЕ на МОН-структурах набули поширення, особливо у складі ВІС, у зв'язку з їх технологічністю, високим ступенем ін-

теграції, низькою вартістю, малою споживаною потужністю та високою навантажувальною здатністю.

БЕ МОН-транзисторної логіки (МОНТЛ) складаються з транзисторів одного типу провідності; для визначеності розглядатимемо БЕ з каналами типу n (n -МОНТЛ), які за логічними рівняннями можуть бути сумісними з БЕ ТТЛ.

Найпростішим і поширеним є інвертор МОНТЛ з *нелінійним навантаженням* (рис. 7.14, *а*), в якому функцію резистора навантаження звичайного ключа виконує транзистор VT_1 , увімкнений за схемою двополюсника, тому що його затвор з'єднано зі стоком, а вхідна напруга надходить до входу керувального транзистора VT_2 . Для транзистора VT_1 завжди виконується співвідношення

$$u_{3.B1} = u_{c.B1} = E_{ж} - u_y,$$

де $u_{3.B1}$, $u_{c.B1}$ – напруги відповідно між заслоном і витоком або стоком і витоком; u_y – вихідна напруга.

У *вимкненому* стані при вхідній напрузі $u_x = U^0 < U_{30}$ транзистор VT_2 закритий, що еквівалентно розімкненому положенню перемикача S (рис. 7.14, *б*), тому стоковий струм не протікає: $i_c = 0$, тобто транзистор VT_1 також закритий. Отже, при цьому його напруги $u_{3.B1} = u_{c.B1} \leq U_{30}$, а напруга на виході перебуває в межах $u_y = E_{ж} \dots U_{30}$ залежно від розкиду опорів закритих транзисторів VT_1 та VT_2 . За рівень логічної одиниці доцільно прийняти мінімальну напругу $U^1 = E_{ж} - U_{30}$, що відповідає робочій точці 1 на вихідних характеристиках транзистора VT_2 (рис. 7.14, *в*) і на передатній характеристиці інвертора (рис. 7.14, *з*).

Вмикання починається, коли зі збільшенням вхідної напруги $u_x \geq U_{30}$ транзистор VT_2 відкривається, напруга на виході зменшується: $u_y \leq E_{ж} - U_{30}$, а на затворі відносно витоку транзистора VT_1 збільшується: $u_{3.B1} \geq U_{30}$ і з'являється струм $i_c > 0$. Лінія навантаження (на рис. 7.14, *в* суцільна лінія R_H) виходить від точки 1 і є нелінійною, тому що за формою вона повторює пунктирну криву $u_c = u_3 - U_{30}$ на ВАХ транзистора VT_1 (рис. 7.12, *в*).

При досягненні вхідною напругою рівня лог. 1 $u_x = U^1$ робоча точка на вихідних і передатній характеристиках опиняється в положенні 2, що відповідає *ввімкненому* стану перемикача S (див. рис. 7.14, б). Для забезпечення низького рівня вихідної напруги $u = U^0 < U_{30}$ аналогічно ключу з лінійним навантаженням необхідно виконати співвідношення між опорами відкритих транзисторів: $R_1 \gg R_2$.

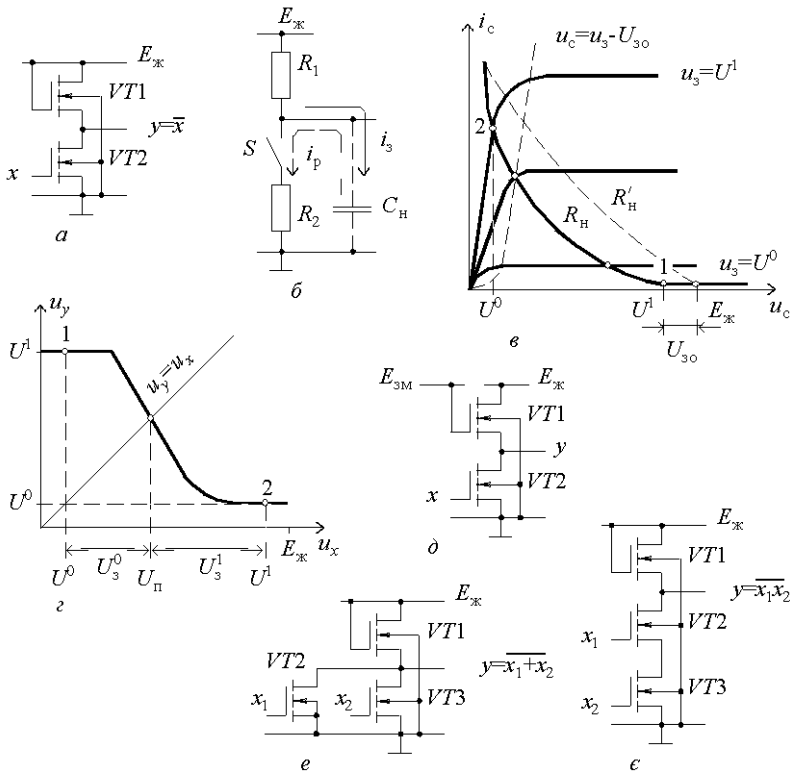


Рис. 7.14. Особливості реалізації базового елемента МОПТЛ

Як порогову можна прийняти напругу $U_{\text{п}}$ за перетином передатної характеристики з лінією $u_y = u_x$, тому що в точці перетину коефіцієнт підсилення великий, особливо в колі інверто-

рів, і незначна зміна вхідної напруги відносно U_{Π} спричиняє перемикання ключів. Завадостійкість $U^0_3=U_{\Pi}-U^0$, $U^1_3=U^1-U_{\Pi}$ при $U_{\Pi} \approx (U^1+U^0)/2$ сягає величини $U^0_3=U^1_3=(U^1-U^0)/2$ і з урахуванням розкиду параметрів та дестабілізувальних чинників становить $U^0_3 \approx U^1_3 \geq 1$ В.

Перехідні процеси, спричинені швидким розрядом навантажувальної ємності C_{Π} через відносно малий опір R_2 і повільним зарядом її через великий опір R_1 , протікають аналогічно ключу з лінійним навантаженням. Підвищення швидкодії шляхом зменшення опорів транзисторів обмежено конструктивно-технологічною спроможністю збільшення їх крутизни.

З огляду на те, що витік навантажувального транзистора VT1 не з'єднано зі спільним для обох транзисторів підшарком, його порогова напруга U_{30} збільшується з підвищенням напруги між витокom і підшарком, що призводить до зменшення вихідної напруги рівня лог. 1, тому що $U^1=E_{\text{ж}}-U_{30}$. Цей недолік усувається в модифікації БЕ з *квазілінійним навантаженням* (рис. 7.14, д): через велику напругу зміщення $E_{3\text{м}} > E_{\text{ж}} + U_{30}$ напруга між затвором і витокom транзистора VT1 завжди більша порогової, тому навантажувальна лінія R'_H (пунктир на рис. 7.14, в) виходить з точки $u_c=E_{\text{ж}}$ і рівень лог. 1 дорівнює $E_{\text{ж}}$. Крім того, внаслідок збільшення струму під час заряду ємності C_{Π} підвищується швидкодія. Проте така модифікація потребує додаткового джерела живлення $E_{3\text{м}}$ та окремих відводів від затворів, що збільшує площу елемента.

Логічна *функція АБО-НІ* реалізується паралельним з'єднанням керувальних транзисторів (рис. 7.14, е), а *функція І-НІ* – їх послідовним з'єднанням (рис. 7.14, є). Коефіцієнт об'єднання входів в останній схемі обмежується зазвичай до $K_{\text{об}} \leq 4$ збільшенням вихідного рівня лог. 0 внаслідок зростання опору між виходом і землею при відкритих усіх керувальних транзисторах. Комбінуванням послідовних і паралельних з'єднань керуваль-

них транзисторів реалізуються складніші функції, наприклад, І-АБО-НІ тощо.

7.3.2. Базовий елемент КМОНТЛ

Базові елементи комплементарної МОН-транзисторної логіки (КМОНТЛ) складаються з транзисторів, що мають канали протилежного (доповняльного) типу провідності (рис.7.15, *а*), причому обидва транзистори: $VT1$ – навантажувальний типу p -МОН та $VT2$ – керувальний типу n -МОН є активними, керуються сигналом x і виконуються з приблизно однаковими параметрами. БЕ працює як двотактний каскад: транзистори $VT1$, $VT2$ по черзі вмикаються і вимикаються, як умовно показано за допомогою перемикачів S_1 , S_2 на рис. 7.15, *б*. Тому що завжди виконується рівність $u_{c.в1} + u_{c.в2} = E_{ж}$, на вихідних характеристиках транзистора $VT2$ (рис. 7.15, *в*) навантажувальні лінії – характеристики транзистора $VT1$ – виходять з точки ($i_c=0$, $u_{c.в}=E_{ж}$).

У *вимкненому статичному стані* інвертора (перемикачі S_1 , S_2 у положенні 1) при $x=0$ напругою низького рівня $U^0 < U_{30}$ транзистор $VT2$ закритий, що відповідає його нижній вихідній характеристиці, знятій при $u_{3,2}=U^0$ (на рис. 7.16, *в* для наочності її ординату збільшено). При цьому між затвором та витоком навантажувального транзистора $VT1$ типу p -МОН діє велика негативна напруга $u_{3,в1}=U^0-E_{ж} \approx -E_{ж}$, що перевищує за модулем порогову: $|u_{3,в1}| > |U_{30}|$. Внаслідок цього транзистор $VT1$ відкритий, чому відповідає його верхня характеристика, знята при $u_{3,в1} \approx -E_{ж}$. Перетин зазначених характеристик у точці 1 визначає вихідну напругу $U^1 \approx E_{ж}$ рівня лог. 1. Аналогічно у *ввімкненому стані* (перемикачі S_1 , S_2 у положенні 2) при $x=1$ рівнем $U^1 > U_{30}$ транзистор $VT2$ відкритий, чому відповідає його верхня характеристика $u_{3,2}=U^1$, а транзистор $VT1$ закритий, тому що між його затвором та витоком напруга $u_{3,в1}=U^1-E_{ж} \approx 0$ менша за порогову

і, отже, його характеристика нижня, знята при $u_{3.B1} \approx 0$. Перетин цих характеристик у точці 2 вказує вихідну напругу $U^0 \approx 0$ рівня лог. 0.

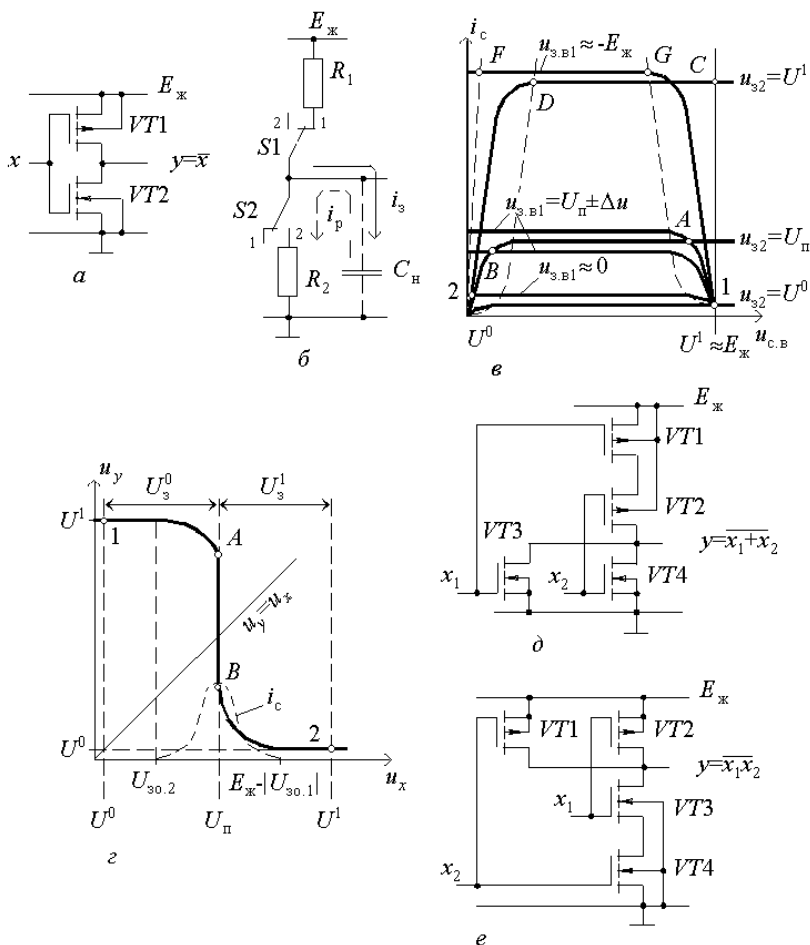


Рис. 7.15. Особливості реалізації базового елемента КМОПТЛ

На передатній характеристиці (рис. 7.15, з) діапазону вхідних напруг $u_x = U^0 \dots U_{0.2}$ відповідає рівень $u_y = U^1$, де транзистор VT2 закритий, а діапазону $u_x = E_{ж} - |U_{30.1}| \dots U_{30.1}$ – рівень $u_y = U^0$,

де закритий транзистор $VT1$. Тому на цих ділянках наскрізний стоковий струм i_c відсутній. На навколопороговій ділянці при $u_x \approx U_{\Pi}$ робочі точки обох транзисторів потрапляють на положисті частини їх вихідних характеристик, тому досить незначної зміни вхідної напруги на $\Delta u \approx 0$ для стрибка точки перетину від A до B , що відповідає майже вертикальній ділянці AB передатної характеристики. При цьому обидва транзистори виявляються відкритими, тому через них протікає наскрізний стоковий струм i_c (пунктир на рис. 7.15, з).

За майже однакових параметрів обох транзисторів порогова напруга БЕ $U_{\Pi} = (U^0 + U^1)/2 \approx E_{\text{ж}}/2$, тому завадостійкість $U^0_3 \approx U^1_3 \approx U_{\Pi}$ виявляється досить високою і з урахуванням розкиду параметрів становить не нижче за $U^0_3 = U^1_3$, тобто 0,9 В. Відсутність споживаного струму $i_c \approx 0$ та споживаної потужності $P_{\text{ж}} \approx 0$ в обох статичних станах, а також великий вхідний опір елементу зумовлюють його високу навантажувальну здатність: $K_{\text{роз}} \approx 100$.

Перехідні процеси, спричинені перезарядом навантажувальної ємності $C_{\text{н}}$, протікають аналогічно БЕ МОНТЛ. Відмінність полягає в тому, що завдяки двотактному режиму роботи вихідні рівні не залежать від опорів відкритих транзисторів $VT1$, $VT2$, тому їх виконують однаковими, з високою крутизною, а, отже, з малим опором, що підвищує навантажувальну здатність і швидкодію. Слід відзначити, що внаслідок перезаряду ємності $C_{\text{н}}$ протягом перемикання БЕ споживаний струм i_c зростає, тому зі збільшенням частоти вхідного сигналу зростає й споживана потужність.

При реалізації логічних функцій у БЕ КМОНТЛ кожному інформаційному входу x_i має відповідати своя пара транзисторів типу p -МОН та n -МОН, аби в статичних станах БЕ не споживали потужність. В елементі *АБО-НІ* (рис. 7.15, д) транзистори n -МОН з'єднують паралельно, а p -МОН – послідовно. При $x_1 = x_2 = 0$ закриті транзистори $VT3$, $VT4$ від'єднують вихід від землі, а відкриті $VT1$, $VT2$ – з'єднують його з джерелом $E_{\text{ж}}$, тому

$y=1$; за будь-яких інших комбінацій вхідних рівнів хоча б один з транзисторів $VT3$, $VT4$ відкритий і з'єднує вихід з землею, а хоча б один з транзисторів $VT1$, $VT2$ закритий і від'єднує вихід від джерела $E_{\text{ж}}$, тому $y=0$. В елементі *I-НІ* (рис. 7.15, *е*), навпаки, транзистори *n*-МОН з'єднано послідовно, а транзистори *p*-МОН – паралельно відносно виходу y . Лише при $x_1=x_2=1$ транзистори $VT3$, $VT4$ відкриті, а $VT1$, $VT2$ закриті, тому $y=0$. При всіх інших комбінаціях хоча б один з транзисторів $VT3$, $VT4$ закритий, а $VT1$, $VT2$ – відкриті, отже, $y=1$.

Слід відзначити, що БЕ КМОНТЛ не можна з'єднувати виходами, тому що при різних станах таких елементів може виникнути наскрізний струм аналогічно БЕ ТТЛ (див. рис. 7.7, *а*). До спільної лінії приєднують модифіковані БЕ КМОНТЛ з трьома станами виходу за умови активізації в певний час керувальним сигналом *OE* лише одного з них. Невикористовувані входи БЕ КМОНТЛ не дозволяється залишати вільними, тому що внаслідок дуже великих вхідних опорів можуть наводитися неприпустимі завади й паразитні потенціали. Такі входи з'єднують з константами, що не змінюють логічну функцію, тобто з "з'ємлем" (лог. 0) або з шиною джерела живлення (лог. 1).

7.3.3. Двоспрямовані ключі КМОНТЛ

На основі КМОН-структур будуються також двоспрямовані ключі з цифровим керуванням, що можуть використовуватися для комутування кіл з аналоговими сигналами. Такий ключ (рис. 7.16, *а, б*) складається з двох увімкнених паралельно транзисторів: $VT1$ типу *p* та $VT2$ типу *n*, підшарки яких аналогічно КМОНТЛ з'єднані відповідно з джерелом живлення $E_{\text{ж}}$ та спільною шиною. Керується ключ цифровим сигналом дозволу *OE*, який за допомогою звичайного інвертора КМОНТЛ (див. рис. 7.15, *а*) надходить до затворів парафазно: OE і \overline{OE} .

При $OE=0$ еквівалентний перемикач *S* (рис. 7.16, *в*) перебуває в розімкненому стані: транзистор $VT2$ типу *n* закритий, тому що у всьому діапазоні вхідних сигналів $u_x=U^0 \dots \approx 0 \dots \approx U_{\text{пруга}}$ між його затвором та витоком $u_{\text{з.в2}} \leq 0$, а транзистор $VT1$

типу p , на затворі якого діє рівень лог. 1, при цьому також закритий позитивною напругою $u_{3.B1} \leq 0$. При $OE=1$ перемикач S замикається: в діапазоні вхідних сигналів $u_x=0 \dots E_{\text{ж}} - U_{30}$) напругою $u_{3.B2} \approx E_{\text{ж}} \dots U_{30}$ відкрито транзистор $VT2$, а в діапазоні $u_x = U_{30} \dots E_{\text{ж}}$ напругою $u_{3.B1} \approx -U_{30} \dots E_{\text{ж}}$ відкрито також транзистор

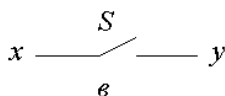
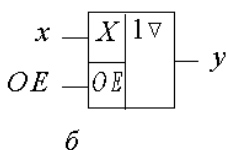
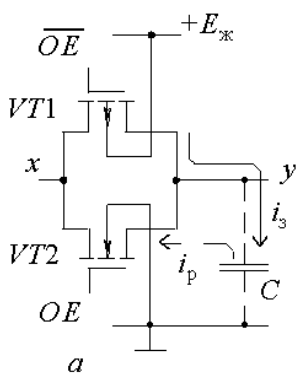


Рис. 7.16. Двоспрямований ключ КМОПТЛ

$VT1$. Отже, аналогові сигнали в діапазоні $u_x = U_{30} \dots E_{\text{ж}} - U_{30}$) передаються з входу до виходу з мінімальною похибкою (тим меншою, чим більший опір навантаження), тому що в цьому діапазоні відкриті обидва транзистори і опір між входом і виходом стає майже постійним.

Слід зауважити, що ключ такого типу є двоспрямований, тому що в МОН-транзисторах симетричної структури електроди стік і витік взаємозамінні: вивід n -каналу з більш високим потенціалом використовується як стік, а з меншим потенціалом – як витік. Внаслідок цього струм через ключ може протікати в обох напрямках: при $x=1$ з надходженням сигналу дозволу $OE=1$ паразитна ємність C заряджається струмом i_3 , при $OE=0$ на ємності певний час зберігається заряд i , нарешті, при $x=0$ та $OE=1$ ємність розряджається струмом i_p у

зворотному напрямку.

7.3.4. Динамічні елементи

На відміну від розглянутих статичних базових елементів, динамічні елементи більшу частину часу перебувають в режимі зберігання інформації, коли вони відімкнені від джерела живлення. Це дає змогу зменшити споживану потужність, а також

підвищити швидкодію та ступінь інтеграції. Роль запам'ятовувального елементу відіграє паразитна сумарна ємність C , здатна досить тривалий час (до мікросекунд) зберігати заряд завдяки великому опору закритих транзисторів елемента і вхідних опорів елементів-навантажень. Аби уникнути втрат інформації внаслідок повільного розряду ємності, її заряд періодично відновлюють протягом дії синхроімпульсів Φ з періодом, не більшим за час зберігання заряду.

Один з варіантів *однотактного* динамічного інвертора (рис. 7.17, а) складається з елементу МОНТЛ на транзисторах $VT1$, $VT2$ (див. рис. 7.14, а) та ключа $VT3$, за допомогою якого запам'ятовувальна ємність C періодично підключається до виходу елементу МОНТЛ. Для цього затвор транзистора $VT1$ відключений від джерела живлення і разом із затвором ключа $VT3$ підключений до джерела синхроімпульсів Φ ; підшарки всіх транзисторів з'єднано з корпусом.

У проміжках між синхроімпульсами Φ (рис. 7.17, б) елемент перебуває в режимі зберігання інформації: закритим транзистором $VT3$ конденсатор C відключений від елемента і він практично зберігає свій заряд, повільно розряджаючись через великі опори втрат. З надходженням чергового імпульсу Φ відбувається відновлення або запис інформації. Так, при $x=0$ транзистор $VT2$ зачинено, імпульсом $\Phi=1$ транзистори $VT1$ та $VT3$ відкриваються, тому конденсатор C дозаряджається струмом i_3 від джерела живлення (під час дії імпульсу 1), якщо він був заряджений, або заряджається (протягом імпульсу 4), якщо був розряджений, тобто при $x=0$ після чергового імпульсу Φ на виході діє рівень $y=1$. І, навпаки, при $x=1$ транзистор $VT2$ відкривається, тому з надходженням синхроімпульсу Φ конденсатор C розряджається струмом i_p (імпульс 2) через малий опір транзистора $VT2$, якщо він був заряджений, або залишається в розрядженому стані (імпульс 3), тобто при $x=1$ на виході буде рівень $y=0$.

Таким чином, в паузах між імпульсами Φ елемент не споживає потужність; відносно входу x він виконує логічну функцію $y=x$. Знак затримки на умовному графічному позначенні (рис. 7.17, в) вказує, що запис інформації відбувається після на-

дходження чергового синхроімпульсу. Для отримання логічних елементів АБО-НІ та І-НІ досить керувальні транзистори $VT2$ з'єднати так само, як і в схемі МОНТЛ (див. рис. 7.14, е, є).

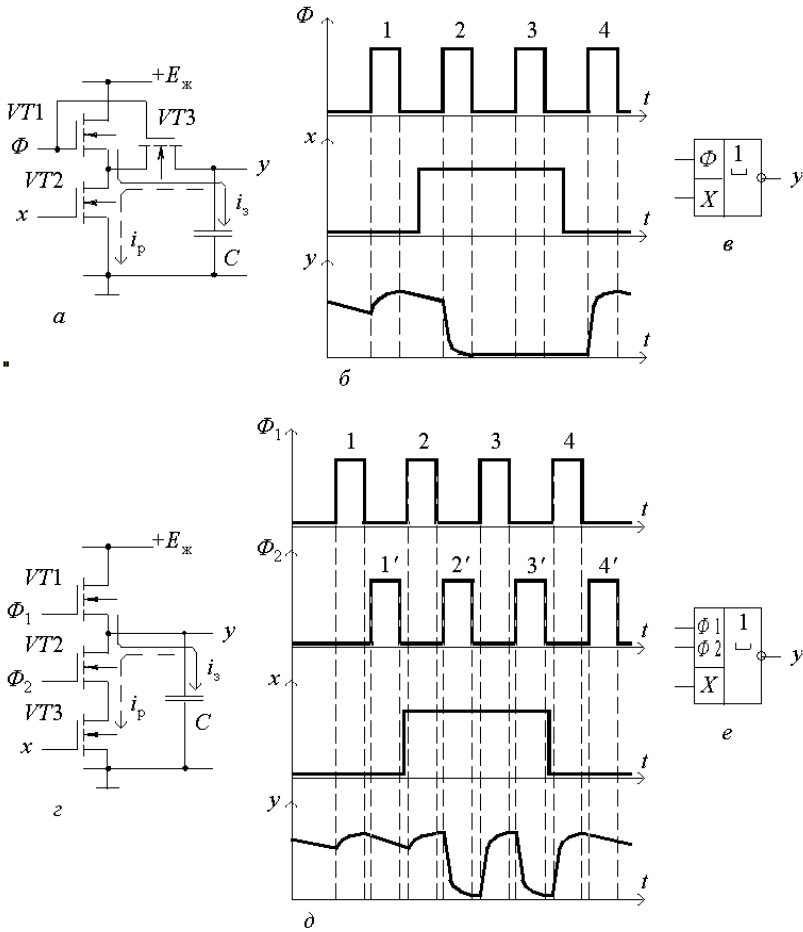


Рис. 7.17. Особливості реалізації динамічних елементів

З огляду на те, що при $\Phi=1$ та $x=1$ відкриваються обидва транзистори $VT1$ і $VT2$, для отримання низького рівня U^0 необхідно, як і в статичній схемі, вибирати відношення опорів цих транзисторів великим, тому таку схему часто називають “з відношенням”. Недоліки елементів “з відношенням” – великий час

заряду ємності C через транзистор $VT1$, що обмежує тривалість імпульсів Φ за мінімумом, та збільшення площі, потрібної для елементу з різними опорами транзисторів, – усуваються в елементах “без відношення”.

Одна з модифікацій *двотактного* елементу (рис. 7.17, *г*) будується на МОН-транзисторах, підшарки яких з’єднано зі спільною шиною, і керується двома зсунутими за фазою послідовностями синхроімпульсів Φ_1 , Φ_2 (рис. 7.17, *д*). У проміжках між синхроімпульсами транзистори $VT1$, $VT2$ закриті, тому конденсатор C перебуває в режимі зберігання заряду. З надходженням синхроімпульсів Φ_1 відкривається транзистор $VT1$ і конденсатор дозаряджається струмом i_z , якщо перед цим він був заряджений (імпульси 1, 2), або заряджається, якщо був розряджений (імпульси 3, 4), тобто по закінченні імпульсу Φ_1 незалежно від значення x на виході завжди діє високий рівень $y=1$.

І, нарешті, з надходженням синхроімпульсів Φ_2 відкривається транзистор $VT2$ і елемент працює в режимі запису інформації: при $x=0$ транзистор $VT3$ закритий, на конденсаторі зберігається заряд (імпульси 1’, 4’) а при $x=1$ транзистор $VT3$ відкривається і конденсатор розряджається через нього струмом i_p (імпульси 2’, 3’) Елемент виконує, таким чином, функцію НІ (рис. 7.17, *е*), а для реалізації функцій АБО-НІ, І-НІ, необхідно, як і в попередньому випадку, з’єднати паралельно чи послідовно керувальні транзистори $VT3$. Недолік динамічних елементів – складність керування – не є вирішальним, особливо для ВІС з внутрішнім пристроєм регенерування. З точки зору користувача такі елементи, по суті, не відрізняються від статичних, проте мають переваги: великий ступінь інтеграції та малу споживану потужність.

7.4. Базові елементи I^2L

В елементах інтегрованої інжекційної логіки (I^2L) можливості схемотехніки, електроніки та технології інтегровані таким чином, що немає чіткої межі між окремими компонентами схеми. Це сприяє підвищенню ступеня інтеграції ВІС на базі I^2L .

Елемент I^2L (на рис. 7.18, *a* відокремлений пунктиром) моделюється схемою з двох транзисторів: керувального $VT1$ та джерела струму I (інжектора) $VT2$. У вимкненому стані БЕ транзистор – джерело сигналу $VT1'$ перебуває в режимі насичення, інформаційному значенню $x=0$ відповідає напруга на його колекторі $U^0=U_{к.н.}$, тому транзистор $VT1$ закритий і струм $I=i_{к1}'$ інжектора $VT2$ замикається до колектора джерела $VT1'$ (на рис. 7.18, *a* струми в цьому режимі зазначені суцільними стрілками), а струм $I=i_{б1}''$ інжектора навантаження $VT2''$ замикається до бази навантажувального транзистора $VT1''$, забезпечує його насичення. При невеликому струмі вихідному значенню $y=1$ відповідає напруга $U^1=U_{пр} \approx 0,7$ В. На передатній характеристиці (рис. 7.18, *б*) вимкнений стан БЕ відображається точкою 1.

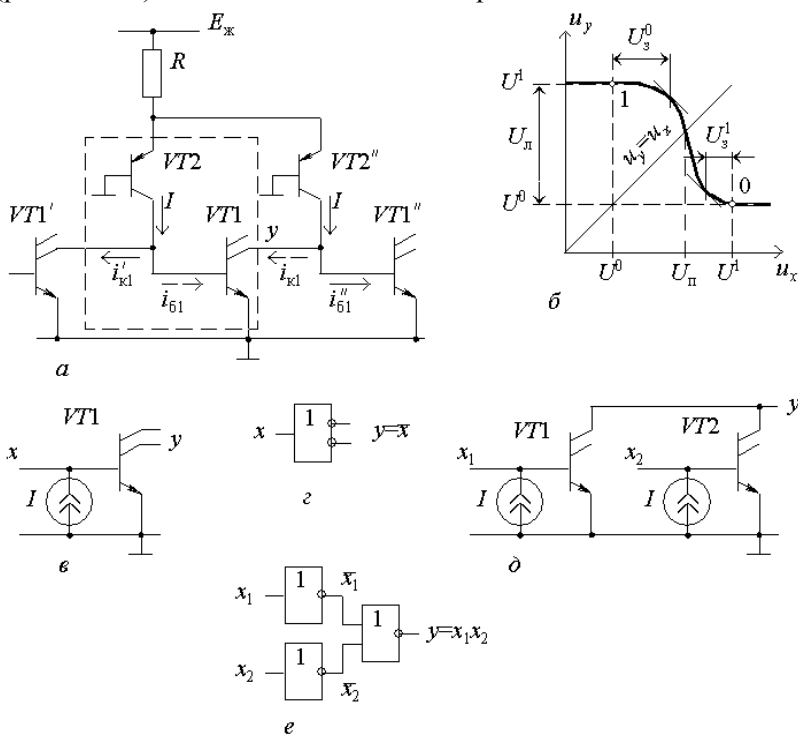


Рис. 7.18. Схеми і характеристики базового елемента I^2L

Увімкнений стан забезпечується, коли транзистор $VT1'$ 'акритий: струм інжектора $I=i_{б1}$ спрямовується до бази транзистора $VT1$ (у цьому режимі струми зазначено пунктиром) і підтримує його в насиченні, тобто значенню $x=1$ відповідає вхідна напруга $U^1=U_{пр} \approx 0,7$ В. Струм $I=i_{к1}$ від БЕ-навантаження тепер замикається до колектора транзистора $VT1$, а вихідною напругою при $y=0$ рівня $U^0=U_{к.н}$ (точка 0 на передатній характеристиці) транзистор $VT1'$ закривається. Тому що при цьому струми $i_{б1}=I$ та $i_{к1}=I$ приблизно однакові, то для забезпечення режиму насичення $i_{б1}>I_{б.н}=I_{к.н}/h_{21e}$ достатній коефіцієнт підсилення $h_{21e} \geq 1$, який можна отримати навіть при мікрострумах, тобто режим увімкнення забезпечується при мінімальній споживаній потужності.

Через малі рівні напруг завадостійкість елементу визначають не відносно порогової напруги $U_{п}$ за перетином лінії одиничного підсилення $u_y=u_x$ з передатною характеристикою, а відносно точок, що відповідають межах відкритого і закритого станів, де $du_y/du_x=1$. Внаслідок різко несиметричної передатної характеристики відносно точок 1 і 0 рівень відкривальної статичної завади $U^0_3 \approx U_{б0}-U^0 \approx U_{л}$ наближається до логічного перепаду $U_{л}=U^1-U^0=U_{пр}-U_{к.н} \approx (0,6...0,7)$ В, а закривальної завади $U^1_3 \approx (20...50)$ мВ дуже малий. Проте через закритий попередній елемент закривальна завада майже не проходить, а рівень динамічних завад, пропорційний логічному перепаду, малий, тому завадостійкість БЕ І²Л є прийнятною для побудови ВІС.

Якщо до виходу y паралельно підімкнути n навантажень, умову насичення $h_{21e} > n$ виконати важче та через розкид вхідних характеристик навантажувальних транзисторів виникає нерівномірний розподіл базових струмів під час їх відкривання. Ці недоліки усуваються за допомогою багатоколекторного транзистора (БКТ): кожне навантаження живиться при цьому від окремого колектора, тому бази навантажувальних транзисторів розв'язані. На еквівалентній схемі (рис. 7.18, в) джерело інжекційного стру-

му VT2 зображається генератором струму I , а рівноцінні й незалежні виходи y на умовному графічному позначенні інвертора (рис. 7.18, з) подають окремими виводами.

При *реалізації логічних функцій* базовим є розглянутий елемент НІ, а також АБО-НІ, який утворюється паралельним з'єднанням колекторів n - p - n -транзисторів (рис. 7.18, д). Якщо хоча б на одному з входів діє рівень лог. 1, відповідний транзистор відкривається, струм навантаження замикається через нього, тому $y=0$. Лише при всіх вхідних рівнях лог. 0 n - p - n -транзистори елементу закриваються і струм навантаження замикається до бази БЕ-навантаження, тобто $y=1$.

Інші логічні функції реалізуються шляхом колекторних об'єднань за формулами алгебри логіки при переході з базису АБО-НІ. Наприклад, функцію І здійснюють на підставі закону де Моргана $\overline{x_1 + x_2} = \overline{x_1} \cdot \overline{x_2}$ (рис. 7.18, е) за допомогою двох елементів НІ та одного АБО-НІ. Необхідно тільки кожне колекторне об'єднання підмикати до окремого навантаження, аби уникнути змагань між базовими струмами навантажувальних транзисторів.

Таким чином, завдяки відсутності в структурі І²Л ізолюючих областей та мінімуму металевих міжз'єднань покращується технологічність виготовлення ІС, зростає ступінь інтеграції, зменшуються паразитні ємності, що разом з відсутністю в схемі резисторів спричиняє зменшення сталих часу ємності і, отже, підвищення швидкодії. Якщо джерело живлення $E_{\text{ж}}$ підключити безпосередньо до елементу, тобто до емітерного переходу транзистора VT2 (рис. 7.18, а), то можна використовувати джерело з низькою напругою $E_{\text{ж}}=U_{\text{пр}}$. Проте для стабілізації інжектованого струму вмикають додатковий резистор R (зазвичай поза ІС) і застосовують напругу живлення $E_{\text{ж}}=(1...5) \text{ В}$. Тому що у включеному стані керованого транзистора його базовий і колекторний струми однакові, елемент може функціонувати при малих струмах (1 $\mu\text{А}$...1 мА), тобто при наднизькій споживаній потужності. Це сприяє поширенню технології І²Л у ВІС, де головний недолік елементів (низька завадостійкість U_3^1) не виявляється.

Запитання та вправи

7.1. Для БЕ типу 1) ТТЛ, 2) ЕСЛ, 3) МОНТЛ, 4) КМОНТЛ, 5) І²Л виконайте завдання:

а) на основній схемі для обох статичних станів позначте у всіх її точках типові значення напруг;

б) на передатній характеристиці позначте в двох статичних станах вхідні і вихідні логічні рівні та порогову напругу, а також визначіть статичну завадостійкість;

в) поясніть за схемою вплив навантажувальної ємності на швидкодію.

7.2. Наведіть принципові електричні схеми, що реалізують зазначені логічні функції на елементах заданого типу:

1) на модифікованих БЕ ТТЛ: а) \overline{x} , б) $\overline{x_1 + x_2}$, в) $\overline{x_1 x_2 + x_3}$, г) $x_1 x_2 x_3$, д) $x_1 + x_2$, е) $x_1 x_2 + x_3$, є) $x_1 x_2$;

2) на двовходових БЕ ЕСЛ: а) $x_1 + x_2 + x_3$, б) $\overline{(x_1 + x_2)x_3}$, в) $x_3 \overline{(x_1 + x_2)}$, г) $\overline{x_1(x_2 + x_3)}$, д) $\overline{x_1 \oplus x_2}$;

3) на БЕ МОНТЛ: а) $\overline{x_1 x_2 x_3}$, б) $\overline{x_1 x_2 x_3}$, в) $\overline{x_1 x_2 + x_3 x_4}$, г) $x_1 \oplus x_2$;

4) на БЕ КМОНТЛ: а) $\overline{x_1 + x_2 + x_3}$, б) $\overline{x_1 + x_2 + x_3}$, в) $\overline{x_1 + x_2 x_3}$, г) $\overline{x_1 x_2}$;

5) на БЕ І²Л: а) $x_1 + x_2 + x_3$, б) $\overline{x_1 + x_2 + x_3}$, в) $x_1 x_2 x_3$, г) $x_1 x_2 x_3$, д) $\overline{x_1 + x_2 x_3}$;

6) на однотактних динамічних елементах МОНТЛ: а) $\overline{x_1 + x_2 + x_3}$, б) $\overline{x_1 x_2 x_3}$, в) $\overline{x_1 x_2 + x_3 x_4}$;

7) на двотактних динамічних елементах МОНТЛ: а) $\overline{x_1 + x_2}$, б) $\overline{x_1 x_2}$, в) $\overline{x_1 x_2 + x_3}$.

7.3. Чому не можна з'єднувати виходи елементів ТТЛ і КМОНТЛ за базовою схемою на спільну лінію зв'язку? Виходи яких модифікацій можна об'єднати на спільне навантаження?

7.4. За допомогою яких засобів у модифікованих елементах ТТЛ досягають:

- а) підвищення навантажувальної здатності;
- б) підвищення швидкодії;
- в) зменшення споживаної потужності;
- г) підвищення завадостійкості;
- д) захисту від неприпустимих вхідних напруг негативної полярності;
- е) можливість об'єднання виходів кількох елементів-передавачів на спільну лінію?

7.5. Якими причинами зумовлена висока швидкодія БЕ ЕСЛ? В яких пристроях доцільно використовувати БЕ ЕСЛ?

7.6. В якому діапазоні з найменшою похибкою комутуються напруги двоспрямованим аналоговим ключем КМОНТЛ?

7.7. Які переваги має динамічний елемент МОНТЛ “без відношення” перед елементом “з відношенням”?

7.8. Чому до одного виходу БЕ I^2L не можна підключити декілька входів навантажувальних елементів? Як усувається цей недолік?

7.9. Порівняйте основні типи БЕ за такими характеристиками:

- а) швидкодією;
- б) ощадливістю щодо споживаної потужності;
- в) енергією перемикання;
- г) завадостійкістю;
- д) навантажувальною здатністю;
- е) ступенем інтеграції.

8. КОМБІНАЦІЙНІ СХЕМИ

8.1. Дешифратори

8.1.1 Загальні положення

Логічні пристрої діляться на комбінаційні та послідовнісні (автомати з пам'я'тю). Вихідні величини логічних пристроїв комбінаційного типу залежать тільки від поточного значення вхідних величин (аргументів). Після завершення перехідних процесів на виходах логічних пристроїв комбінаційного типу встановлюються вихідні величини, на які характер перехідних процесів впливу не має. До комбінаційних пристроїв належать дешифратори і шифратори, мультиплексори та демультиплексори, а також суматори кодів.

Дешифратори і шифратори належать до числа перетворювачів кодів. З поняттям шифрування зв'язане представлення про стискання даних, а з поняттям дешифрування – зворотне перетворення. В умовних позначеннях дешифраторів і шифраторів використовуються літери DC і CD (від слів decoder і coder відповідно).

Повним дешифратором називається комбінаційна схема (КС), що має n входів і 2^n виходів, яка реалізує на кожному виході функцію, що представляє собою мінтерм n вхідних змінних. У повному дешифраторі кожній комбінації значень вхідних сигналів відповідає сигнал тільки на одному з виходів. Іншими словами, дешифратором називають логічний пристрій, що має n входів та $m = 2^n$ виходів і який перетворює код числа, що надходить на його входи, у позиційний (унітарний) код, тобто сигнал формується на кожному відрізьку часу тільки на одному з його виходів.

Якщо вхідний код двійковий, то функція дешифратора описується такими виразами:

$$y_0 = x_1 x_2 x_3 \dots x_n,$$

$$y_1 = \bar{x}_1 x_2 x_3 \dots x_n,$$

.....

$$y_m = \bar{x}_1 \bar{x}_2 \bar{x}_3 \dots \bar{x}_n,$$

де, n – кількість розрядів вхідного коду; m – можлива кількість виходів дешифратора; x_i – вхідна змінна.

Дешифратори можуть реалізовувати як функції, наведені вище, так й інверсні їм значення. Залежно від цього розрізняють дешифратори з прямими та інверсними виходами (рис. 8.1). При цьому всі дешифратори ТТЛ мають інверсні виходи. Інверсія на виході означає, що на всіх виходах зберігається “1” крім активного виходу, рівень на якому дорівнює “0”.

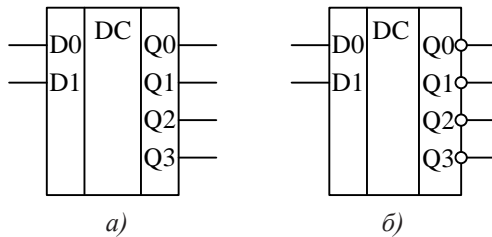


Рис. 8.1. Умовне графічне позначення дешифратора з двома входами адреси і прямими (а) та інверсними виходами (б)

Дешифратор може бути виконаний одноступінчатим (лінійним) і багатоступінчатим. Багатоступінчаті можна розділити на прямокутні (матричні) та пірамідальні.

8.1.2. Лінійні дешифратори

Лінійні дешифратори виконуються реалізацією виразів, наведених вище, напрями. Для побудови лінійного дешифратора потрібно використання логічних елементів з кількістю входів, яка дорівнює розрядності вхідного коду. Крім того, потрібно є висока навантажувальна здатність того пристрою, що формує вхідні коди, тому що до його вихідних шин підключається велика кількість входів логічних елементів, які утворюють дешифратор. Практично навантажувальна здатність не перевищує 10...20.

Лінійний суматор навантажує вхідний пристрій на 2^{n-1} входів. На рис. 8.2 наведена схема лінійного дешифратора на три входи.

8.1.3. Прямокутні (матричні) дешифратори

Удосконалення структури дешифраторів можливо при використанні ряду елементів для формування часткових кон'юнкцій, використовуваних надалі для отримання необхідних вихідних функцій дешифратора.

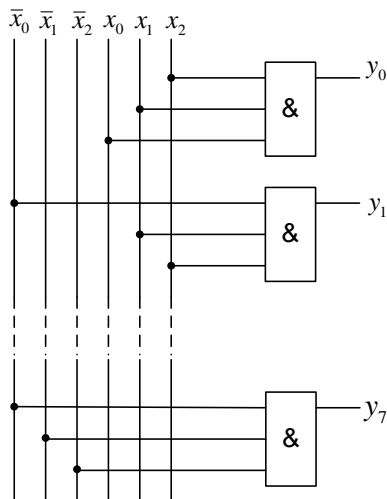


Рис. 8.2. Схема лінійного дешифратора на три входи

Прямокутний дешифратор містить перший ступінь з кількох лінійних дешифраторів, кожних з яких дешифрує групу розрядів вхідного коду. Усі кон'юнкції всередині групи утворюються за допомогою допоміжних дешифраторів, а необхідні вихідні змінні дешифратора (мінтерми) формуються в результаті кон'юнкцій вихідних змінних допоміжних дешифраторів.

Багаторозрядне слово розбивається на N груп. При парному n обидві групи однакові та містять $n/2$ входів кожна. При непарному n перша група містить $(n+1)/2$ входів, а друга група – $(n-1)/2$.

Оцінюючи, як і раніше, навантажувальну здатність вихідних елементів джерела вхідного коду, можна зробити висновок

про необхідність додаткової розбивки груп на підгрупи (якщо навантажувальна здатність виявляється меншою, ніж число входів лінійного дешифратора) або про можливість застосування як допоміжного лінійного дешифратора. Якщо потрібні додаткові розбивки на підгрупи, то допоміжний дешифратор розподіляє за двосхідчастою схемою, а весь дешифратор буде виконаним за трисхідчастою схемою.

Розглянемо приклад побудови прямокутного дешифратора на 4 входи і 16 виходів. Нехай навантажувальна здатність вихідних елементів пристрою, що формують вихідний код дешифратора, дорівнює 3. Тоді дешифратор необхідно будувати за двосхідчастій схемі.

Розбиваємо усі входи на 2 групи по 2 входи в кожній. Перевіряємо, чи буде достатньою навантажувальна здатність для $N = 2: (2^N + 1) / 2 = 2,5$.

Структура дешифратора буде виглядати так, як показано на рис. 8.3.

1 група	$x_0 x_1$	$x_0 x_1 x_2 x_3$	$- y_0$
	$\bar{x}_0 x_1$	$\bar{x}_0 x_1 x_2 x_3$	$- y_1$
	$x_0 \bar{x}_1$	$x_0 \bar{x}_1 x_2 x_3$	$- y_2$
	$\bar{x}_0 \bar{x}_1$	$\bar{x}_0 \bar{x}_1 x_2 x_3$	$- y_3$
		.	.
2 група		.	.
	$x_2 x_3$	$x_0 x_1 \bar{x}_2 \bar{x}_3$	$- y_{12}$
	$\bar{x}_2 x_3$	$\bar{x}_0 x_1 \bar{x}_2 \bar{x}_3$	$- y_{13}$
	$x_2 \bar{x}_3$	$x_0 \bar{x}_1 \bar{x}_2 \bar{x}_3$	$- y_{14}$
	$\bar{x}_2 \bar{x}_3$	$\bar{x}_0 \bar{x}_1 \bar{x}_2 \bar{x}_3$	$- y_{15}$

Рис. 8.3. Структура прямокутного дешифратора

8.1.4. Пірамідальні дешифратори

Пірамідальні дешифратори, так само як і прямокутні, відносяться до розряду багатоступінчастих дешифраторів, особливістю яких є застосування у всіх групах дешифрації двовхо-

дових логічних елементів з обов'язковим підключенням виходу елементу K -го ступеню до входів тільки 2-х елементів $(K+1)$ -го ступеню. Число ступенів K в пірамідальному дешифраторі на одиницю менше розрядності вхідного коду $K = N-1$, а число логічних елементів кожного ступеню визначається за формулою $B_i = 2^{i+1}$, де i – номер ступеню пірамідального дешифратора. На рис. 8.4 наведено приклад побудови трирозрядного дешифратора.

В дешифраторах часто передбачається операція стробування (тактування), що дозволяє формувати вихідні сигнали тільки у визначені стробуючими імпульсами інтервали часу. Стробування може здійснюватися введенням додаткового входу паралельно інформаційним (рис. 8.5, а) до кожного елементу дешифратора або блокуванням всіх елементів через одне з вхідних кіл (рис. 8.5, б).

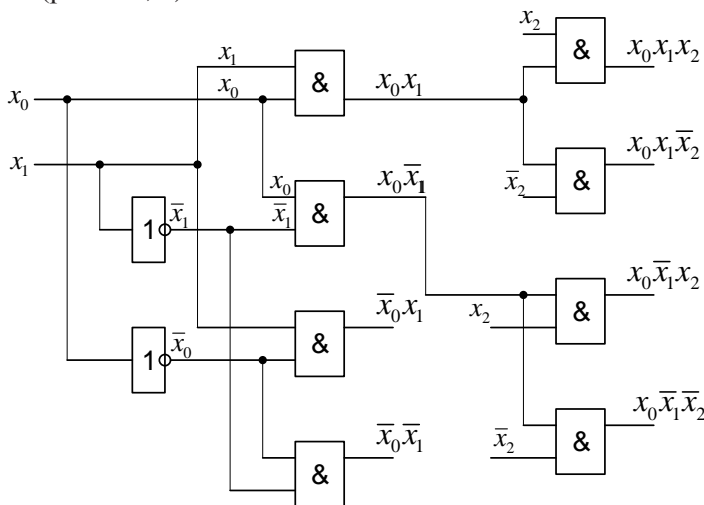


Рис. 8.4. Структура трирозрядного пірамідального дешифратора

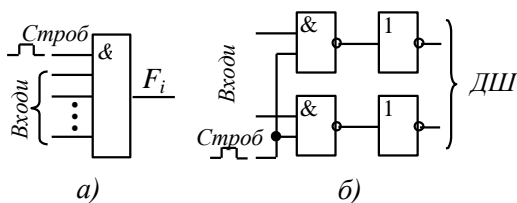


Рис. 8.5. Способи стробування дешифратора

В другому випадку при нульовому значенні сигналу стробування штучно формуються нулі в колах прямого й інверсного значень змінної, що забезпечує наявність хоча б одного нуля в числі вхідних величин для кожного елементу дешифратора.

При цьому усі виходи мають нульові значення сигналу, тому що будь-яка змінна в прямому чи інверсному вигляді надходить на всі елементи дешифратора. При одиничному значенні сигналу стробування відновлюється нормальне коло передачі змінної на входи дешифратора. Описані варіанти можна назвати стробуванням по виходу і стробуванням по входу.

8.2. Шифратори

Шифратором називається комбінаційна схема, що перетворює сигнал, поданий на одну з вхідних шин, у визначену кодову комбінацію на виході (операція, зворотна дешифруванню). У країнах Євросоюзу та США такі мікросхеми позначаються літерами SN (наприклад, SN74148N), а в Росії та країнах митного союзу літерами ИВ (аналог K555ИВ3).

За принципом роботи шифратори діляться на двійкові та пріоритетні. При збудженні одного з входів шифратора на його виході формується двійковий код, що відповідає номеру входу. Повний двійковий шифратор має 2^n входів і n виходів. Відповідність вхідного та вихідного кодів двійкового шифратора можна побачити у табл. 8.1.

Таблиця 8.1. Відповідність вхідного та вихідного кодів шифратора

Стан входу				Код на виході	
D0	D1	D2	D3	Q1	Q0
1	0	0	0	0	0
0	1	0	0	0	1
0	0	1	0	1	0
0	0	0	1	1	1

З таблиці випливає, що $Q0 = \overline{D3D2D1D0} + D3\overline{D2D1D0}$,
а $Q1 = \overline{D3D2D1D0} + D3\overline{D2D1D0}$.

Пріоритетні шифратори виконують більш складні функції. В такому шифраторі кожному з входів присвоюється пріоритет. Найчастіше входу з найменшим номером присвоюється найнижчий пріоритет. У випадку, коли сигнали надходять на декілька входів шифратора, на виході формується код, що відповідає входу з найвищим пріоритетом.

8.3. Мультиплексори

Мультиплексором називається комбінаційна схема (КС), що має $m+2^m$ входів і один вихід, де m – кількість адресних входів, а 2^m – число інформаційних входів мультиплексора.

Мультиплексори (англ. multiplexer, позначається MUX або MX) використовуються в інформаційних та інформаційно-вимірювальних системах для розділу каналів інформації у часі, тобто по черзі підключають різні джерела інформації до каналу зв'язу. Іншими словами, мультиплексор виконує функцію прийому інформації по одному з n вхідних каналів та передає її по єдиному вихідному каналу.

Еквівалентна схема мультиплексора, що має два входи й один вихід, може бути представлена у вигляді, що наведений на рис. 8.6.

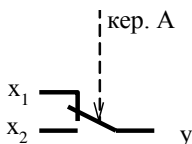


Рис. 8.6. Еквівалентна схема мультимплексора з двома входами

Входи мультимплексора діляться на дві групи: інформаційні та керуючі (адресні). Робота такого мультимплексора описується рівнянням

$$y = x_1 \bar{A} + x_2 A.$$

Якщо $A=0$ (значення сигналу на адресному вході), то перемикач знаходиться в положенні x_1 , а якщо $A = 1$, то перемикач знаходиться у положенні x_2 .

Схема мультимплексора з двома інформаційними входами, одним входом адреси та одним виходом наведена на рис. 8.7.

Позначення 2×1 означає, що мультимплексор має 2 входи і 1 вихід. Приклад умовного позначення мультимплексора наведений на рис. 8.8.

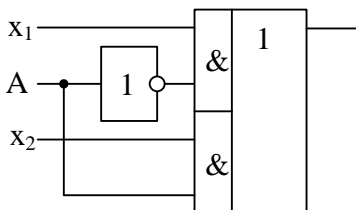


Рис. 8.7. Схема мультимплексора 2×1

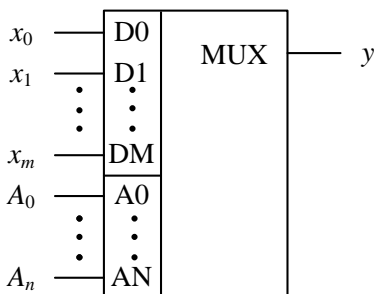


Рис. 8.8. Умовне позначення багатовходового мультимплексора

Якщо побудувати мультимплексор на 8 входів і 1 вихід, то його характеристичне рівняння буде мати вигляд:

$$y = x_0 \bar{A}_0 \bar{A}_1 \bar{A}_2 + x_1 \bar{A}_0 \bar{A}_1 \bar{A}_2 + x_2 \bar{A}_0 \bar{A}_1 \bar{A}_2 + x_3 \bar{A}_0 \bar{A}_1 \bar{A}_2 + \dots + x_7 A_0 A_1 A_2.$$

Оскільки зміна кодів на адресних входах (управління) відповідає рівнянням трирозрядного дешифратора, то функціональну схему мультимплексора можна зобразити так, як показано на рис. 8.9.

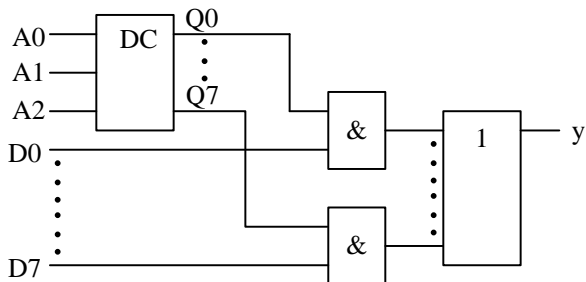


Рис. 8.9. Функціональна схема мультимплексора на 8 входів та 1 вихід

В даний час промисловістю випускаються серії мікросхем, до складу яких входять мультимплексори, що мають кількість адресних входів $m = 1, 2, 3$ та 4. Мікросхеми мультимплексорів, що випускаються в межах СНД, позначаються літерами КП, наприклад КР1533КП2.

Крім мультимплексорів, що мають один вихід, можна організувати мультимплексори, що мають кілька виходів. Наприклад, мультимплексор $2 \times (4 \times 1)$ містить 2 мультимплексори, кожний з яких містить 4 входи і 1 вихід. Умовне графічне позначення зведеного чотириканального мультимплексора наведено на рис. 8.10.

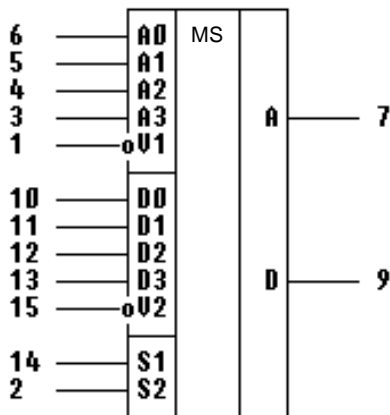


Рис. 8.10. Умовне графічне позначення здвоєного 4-канального мультиплексора

8.4. Демультимплексори

При передаванні даних спільним каналом з поділом у часі потрібні не тільки мультиплексори, але і пристрої зворотного призначення, що розподіляють дані з одного каналу між декількома приймачами інформації. Демультимплексори виконують функцію, зворотну мультиплексорам, тобто проводять комутацію одного інформаційного вхідного сигналу на 2^m виходів, де m – кількість адресних входів. Мультиплексори, що виконані на основі аналогових двонаправлених ключів (наприклад, 564КП1 та 564КП2), можуть виконувати функції і демультимплексорів.

Еквівалентна схема демультимплексора, що має один вхід і два виходи, наведена на рис. 8.11.

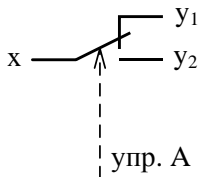


Рис. 8.11. Еквівалентна схема демультимплексора 1×2

Умовне графічне позначення демультимплексора на схе-

мах наведено на рис. 8.12.

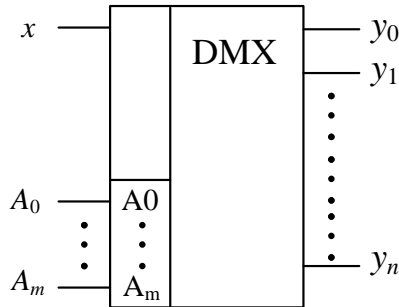


Рис. 8.12. Умовне графічне позначення демультиплексора

Кількість виходів демультиплексора визначається за формулою $n = 2^m$. Характеристичні рівняння демультиплексора мають такий вигляд:

$$y_0 = x \overline{A_0} \overline{A_1} \dots \overline{A_m},$$

$$y_1 = x A_0 \overline{A_1} \dots \overline{A_m},$$

$$y_2 = x A_0 A_1 \dots \overline{A_m},$$

.....,

$$y_n = x A_0 A_1 \dots A_m.$$

При $x=1=\text{const}$ демультиплексор перетворюється на звичайний дешифратор.

Схема демультиплексора 1×4 (1 вхід та 4 виходи) наведена на рис. 8.13.

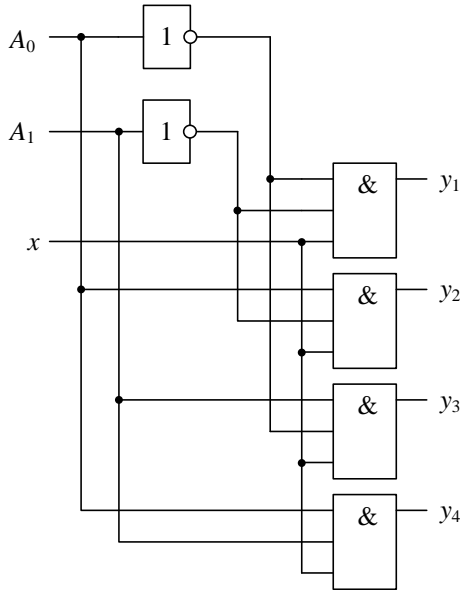


Рис. 8.13. Схема демультиплексора 1×4

8.5. Схемна реалізація логічних функцій на дешифраторах та мультиплексорах

8.5.1. Проектування логічної схеми на дешифраторі

Запишемо МДНФ, знайдену у розділі 6.5, виходячи з якої будемо виконувати проектування схеми:

$$y = \overline{x_4} \overline{x_2} + x_3 x_4 + x_1 \overline{x_2}.$$

1. значаємо залежність функції від тієї або іншої змінної:

$$11. \quad \tilde{x}_1 \rightarrow 1; \quad \tilde{x}_2 \rightarrow 2; \quad \tilde{x}_3 \rightarrow 1; \quad \tilde{x}_4 \rightarrow 2.$$

2. Вибираємо адресні змінні (сигнали), тобто ті, від яких функція залежить найбільше.

Очевидно, що від деяких змінних функція залежить найбільше, тому перевіримо два варіанти реалізації.

$$1) \quad A = x_4 x_2 x_1;$$

$$2) \quad A = x_4 x_2.$$

Тому послідовно розглянемо обидва ці варіанти.

Для першого варіанту опишемо функції, які визначають сигнали на виходах дешифратора залежно від адресного коду.

$$y_0 = \bar{x}_4 \bar{x}_2 \bar{x}_1;$$

$$y_1 = \bar{x}_4 \bar{x}_2 x_1;$$

$$y_2 = \bar{x}_4 x_2 \bar{x}_1;$$

$$y_3 = \bar{x}_4 x_2 x_1;$$

$$y_4 = x_4 \bar{x}_2 \bar{x}_1;$$

$$y_5 = x_4 \bar{x}_2 x_1;$$

$$y_6 = x_4 x_2 \bar{x}_1;$$

$$y_7 = x_4 x_2 x_1;$$

Визначимо, як залежить вихідна функція від сигналів на виходах дешифратора:

$$\begin{aligned} y &= y_0 + y_5 + y_7 + y_6 + y_7 + x_2 x_3 = \\ &= y_0 + y_5 + y_6 + y_7 + x_2 x_3 \end{aligned}$$

Перетворимо отриману функцію до базису І-НІ та реалізуємо схему (рис.8.14):

$$y = \overline{\overline{y_0} \overline{y_5} \overline{y_6} \overline{y_7} \overline{x_2 x_3}}.$$

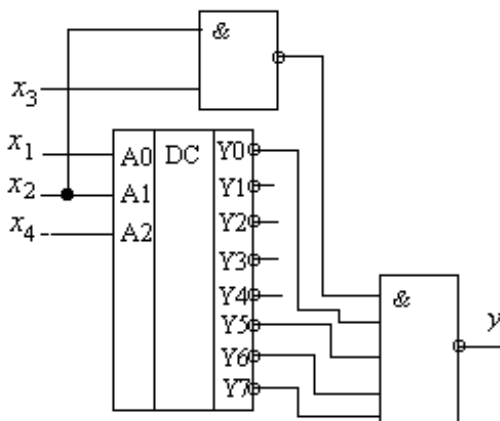


Рис. 8.14. Перший варіант реалізації схеми на дешифраторі

Далі визначимо, як залежить вихідна функція від сигналів на виходах дешифратора для другого варіанту реалізації:

$$y_0 = \bar{x}_4 \bar{x}_2;$$

$$y_1 = \bar{x}_4 x_2;$$

$$y_2 = x_4 \bar{x}_2;$$

$$y_3 = x_4 x_2;$$

$$y = y_3 + x_2 x_3 + x_1 x_4 + x_1 y_0.$$

Перетворимо отриману функцію до базису І-НІ та реалізуємо схему (рис. 8.15):

$$y = \overline{\overline{y_3} \overline{x_2 x_3} \overline{x_1 x_4} \overline{x_1 y_0}}.$$

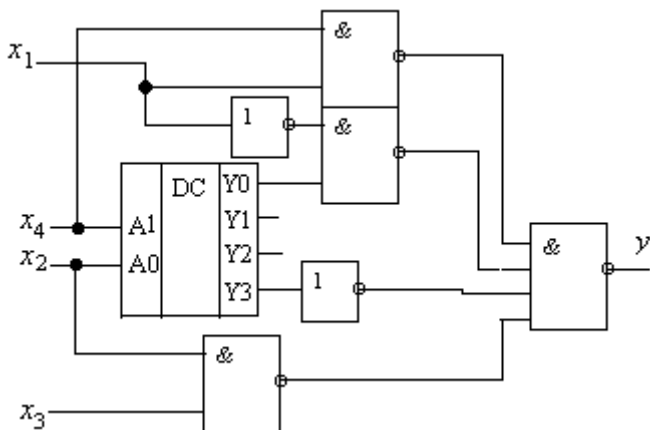


Рис. 8.15. Другий варіант реалізації схеми на дешифраторі

Розглянувши два представлені варіанти, приходимо до висновку, що більш просту реалізацію має перший варіант схеми з комбінацією адресних змінних $A = x_4x_2x_1$.

8.5.2. Проектування логічної схеми на мультиплексорі

x_1			
1			
1		1	1
	1	1	1
	1	1	1
x_2			
x_1			

x_1			
0	1	3	2
0	1	3	2
4	5	7	6
4	5	7	6
x_2			

Для проектування в якості варіантів адресних змінних використаємо ті самі, що й для дешифратора.

$$A = x_4x_2x_1$$

Розіб'ємо діаграму на j піддіаграм, що дорівнює кількості інформаційних входів мультиплексора.

Запишемо десяткове значення коду $j = A = x_4x_2x_1$ у кожній клітинці. З вхідної діаграми зчитуємо інформаційні значення на входах мультиплексора, а з другої діаграми зчитуємо номери піддіаграм. Отже, визначимо логічні функції на інформаційних входах мультиплексора:

$$d_1 = d_4 = 0; \quad d_0 = d_5 = d_6 = d_7 = 1; \\ d_3 = d_2 = x_3.$$

Схема проектованого пристрою наведена на рис. 8.16.

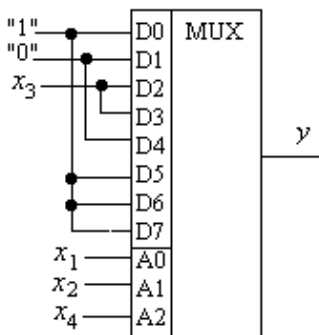


Рис. 8.16. Перший варіант реалізації схеми на мультиплексорі

За другим варіантом розіб'ємо діаграму на j піддіаграм, що дорівнює кількості інформаційних виходів мультиплексора, тобто $j = A = x_4 x_2$, та запишемо десяткові значення цього коду у кожній клітинці на карті Карно.

j	0	0	1	1
	0	0	1	1
x_4	2	2	3	3
	2	2	3	3

x_2

З вхідної діаграми зчитуємо інформаційні значення на входах мультиплексора, а з другої діаграми зчитуємо номери піддіаграм. Отже, визначаємо інформаційні значення на входах мультиплексора: $d_3 = 1$;

$$d_0 = \overline{x_1}$$

$$d_0 \begin{array}{|c|c|} \hline \overline{x_1} & \\ \hline 1 & \\ \hline 1 & \\ \hline \end{array} | x_3$$

$$d_1 \begin{array}{|c|c|} \hline & \overline{x_1} \\ \hline 1 & 1 \\ \hline \end{array} | x_3$$

$$d_1 \begin{array}{|c|c|} \hline & \overline{x_1} \\ \hline & 1 \\ \hline & 1 \\ \hline \end{array} | x_3$$

$$\begin{aligned} d_1 &= x_3 \\ d_2 &= x_1 \end{aligned}$$

В отриманому варіанті реалізації, у порівнянні з попереднім, дещо спрощується мультиплексор (необхідна кількість адресних входів дорівнює 2), однак до складу схеми необхідно додатково ввести інвертор для змінної x_1 .

8.6. Суматори кодів

Суматором називається комбінаційний логічний пристрій, призначений для виконання операції арифметичного додавання чисел, представлених у вигляді двійкових кодів.

Суматори є одним з основних вузлів арифметико-логічного пристрою. Термін суматор охоплює широкий спектр пристроїв, починаючи з найпростіших логічних схем і до складних цифрових вузлів. Спільним для всіх цих пристроїв є арифметичне додавання чисел, представлених у двійковій формі. Класифікація суматорів може бути виконана за різними ознаками.

За кількістю розрядів розрізняють півсуматори, однорозрядні суматори та багаторозрядні суматори.

Півсуматором називається пристрій, призначений для додавання двох однорозрядних слів, який має два входи і два виходи та формує з вхідних сигналів сигнали суми і перенесення до старшого розряду.

Однорозрядним суматором називається пристрій, призначений для додавання двох однорозрядних слів, який має три входи і два виходи та формує з сигналів вхідних доданків і сигналу перенесення з молодших розрядів сигнали суми і перенесення до старшого розряду.

Багаторозрядним суматором називається пристрій, призначений для додавання двох багаторозрядних слів, який формує на виході код суми і сигнал перенесення у випадку, якщо результат додавання не може бути представлений кодом, розрядність якого збігається з розрядністю кодів доданків.

Багаторозрядні суматори діляться на послідовні і паралельні. В послідовних суматорах операція додавання виконується послідовно розряд за розрядом, починаючи з молодшого. В паралельних всі розряди вхідних кодів додаються одночасно.

Розрізняють комбінаційні суматори, тобто пристрої, які не мають власної пам'я'ї, та накопичувальні суматори з власною внутрішньою пам'я'тю, в якій акумулюються результати виконаної операції. При цьому кожний черговий доданок додається до того значення, яке зберігалось у пристрої.

За способом тактування розрізняють синхронні й асинх-

ронні суматори. В синхронних суматорах час виконання операції арифметичного підсумовування двох кодів не залежить від виду самих кодів і завжди залишається постійним. В асинхронних суматорах час виконання операції залежить від виду доданків. Тому по завершенню виконання підсумовування необхідно виробляти спеціальний сигнал завершення операції.

В залежності від використовуваної системи числення розрізняють двійкові, двійково-десяткові та інші типи суматорів.

Визначимо функції, що описують операції арифметичного додавання двох однорозрядних двійкових кодів x_1 та x_0 . Алгоритм її виконання пояснюється таблицею істинності (табл. 8.2). У графі s (сума) наведене значення результату додавання, а в графі p (перенесення) отримане при цьому значення перенесення в старший розряд. Варто звернути увагу на відмінності результатів, отримуваних при арифметичному і логічному додаваннях. При логічному додаванні в останньому рядку стовпця s було би присутнє значення 1, тобто не виникало б перенесення до старшого розряду. Ця відмінність результатів даних операцій не дозволяє застосувати для арифметичного підсумовування елемент АБО, а вимагає розробки спеціалізованого пристрою.

Таблиця 8.2. Формування результату додавання однорозрядних двійкових чисел

x_1	x_0	s	p
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Значення сигналу перенесення, який в останньому рядку таблиці 8.2 дорівнює одиниці, говорить про те, що результат, отриманий при виконанні операції арифметичного додавання, в цьому випадку не може бути представлений двійковим кодом, розрядність якого дорівнює розрядності слів доданків. Для представлення результату необхідне слово, що має на один розряд більше, ніж коди доданків.

Використовуючи приведену таблицю, легко записати си-

стему функцій алгебри логіки, що описують алгоритм операції арифметичного додавання:

$$s = \overline{x_1}x_0 + x_1\overline{x_0} = x_1 \oplus x_0; \quad (8.1)$$

$$p = x_1x_0.$$

Для технічної реалізації схеми півсуматора необхідні логічні елементи І та виключне АБО (рис. 8.17).

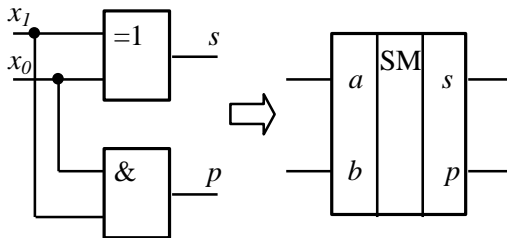


Рис. 8.17. Схема півсуматора та його умовне позначення

Операція додавання додатних двійкових чисел визначається правилами двійкової арифметики:

- значення перенесення z_{p+1} у $(p + 1)$ -ий розряд дорівнює 1, якщо дві або три величини з x_p , y_p та z_p дорівнюють 1, де x_p і y_p – розряди чисел X та Y , а z_p – перенесення з $(p - 1)$ -го розряду;
- значення p -го розряду s_p суми чисел X та Y дорівнює 1, якщо непарне число величин x_p , y_p та z_p дорівнює 1.

Значення перенесення до першого розряду завжди дорівнює 0, тобто $z_1 = 0$. Якщо при додаванні розрядна сітка не переповняється, то перенесення до старшого $(n+1)$ -ого розряду відсутнє ($z_{n+1} = 0$). В загальному випадку необхідно проводити додавання як додатних, так і від’ємних чисел.

Таблиця істинності (табл. 8.3), що описує закон функціонування однорозрядного двійкового суматора, складається на підставі правил додавання додатних чисел.

Один з можливих варіантів реалізації однорозрядного двійкового суматора наведено на рис. 8.18.

Для додавання двох n -розрядних чисел X та Y потрібно використовувати n однорозрядних суматорів. На рис. 8.19 наведена функціональна схема суматора для трирозрядних чисел $X = (x_3, x_2, x_1)$ та $Y = (y_3, y_2, y_1)$.

Таблиця 8.3. Таблиця істинності однорозрядного двійкового суматора

i	x_p	y_p	z_p	s_p	z_{p+1}
0	0	0	0	0	0
1	0	0	1	1	0
2	0	1	0	1	0
3	0	1	1	0	1
4	1	0	0	1	0
5	1	0	1	0	1
6	1	1	0	0	1
7	1	1	1	1	1

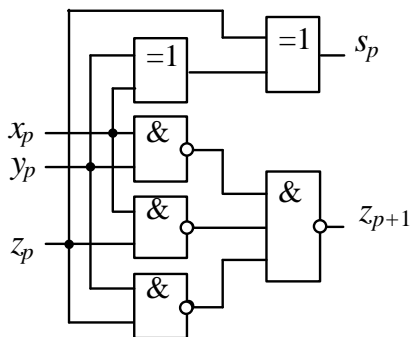


Рис. 8.18. Схема однорозрядного двійкового суматора

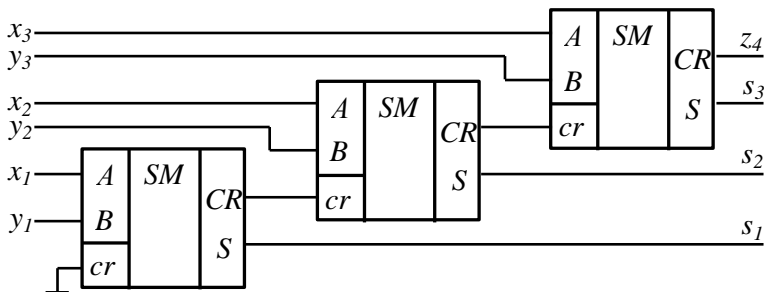


Рис. 8.19. Функціональна схема суматора трирозрядних чисел

8.7. Схеми порівняння кодів або цифрові компаратори

Схемою порівняння кодів або цифровим компаратором називається комбінаційний логічний пристрій, призначений для порівняння чисел, представлених у вигляді двійкових кодів. Число входів компаратора визначається розрядністю порівнюваних кодів.

Цифрові компаратори, що реалізуються у вигляді цифрових схем, формують на виходах три ознаки: $A = B$, $A > B$ та $A < B$, де A і B – значення кодів, що порівнюються.



Рис. 8.20. Функціональні позначення цифрового компаратора

Найбільш поширений спосіб порівняння двох чисел, заздалегідь записаних в регістри, оснований на їх порозрядному порівнянні, починаючи з старшого, тобто використовується принцип послідовного порівняння кодів багаторозрядних чисел. Якщо порівнюються два числа $A(a_n, a_{n-1}, \dots, a_1)$ і $B(b_n, b_{n-1}, \dots, b_1)$, то умовою їх рівнозначності ($A = B$) є рівність кодів усіх одиниць розрядів, а умовою нерівнозначності ($A \neq B$) – нерівність кодів хоча б у одному розряді, при цьому $A > B$, якщо $a_i = 1, b_i = 0$. Звідси булеві функції набувають такого вигляду:

$$Q_{a_i=b_i} = \overline{a_i} \overline{b_i} + a_i b_i,$$

$$Q_{a_i>b_i} = a_i \overline{b_i},$$

$$Q_{a_i<b_i} = \overline{a_i} b_i.$$

На рис. 8.21, а, б показані дві схеми, що реалізують функцію рівності двох кодів $R(v)$.

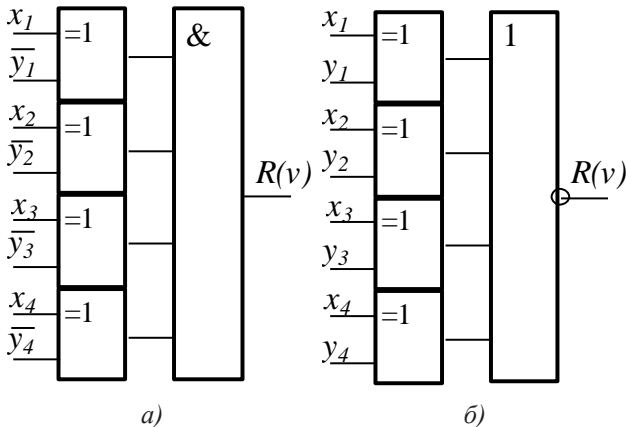


Рис. 8.21. Схеми формування ознаки рівнозначності 4-розрядних кодів

Для того, щоб сформувати ознаки $A > B$ та $A < B$ за допомогою логічних елементів, знадобиться досить складна схема. Найбільш зручно будувати схеми порівняння кодів з використанням суматорів. Якщо коди, що порівнюються, однакові, то різниця їх значень дорівнює 0. Інші дві ознаки теж нескладно сформувати.

Розглянемо на прикладі яким чином можна побудувати схему порівняння кодів з використанням суматора.

Для цього здійснимо операцію віднімання двох 4-розрядних чисел для трьох випадків: $A = B$, $A > B$ та $A < B$. Нагадаємо, що для того, щоб відняти одне число від іншого, необхідно один з доданків представити в доповняльному коді, тобто проінвертувати всі розряди числа і додати 1 до молодшого розряду.

Перший випадок $A = B$. Нехай $A_{10} = B_{10} = 5_{10} = 0101_2$.

Представимо число B в доповняльному коді:

$$1010$$
$$+ 1$$
$$\overline{1011}$$

Тобто сума дорівнює $s = 1011$, а перенесення $p = 0$. Тепер складемо ці два числа:

$$0101$$
$$+1011$$
$$\overline{1\ 0000}$$

Для отриманого результату $s = 0000$, а перенесення $p = 1$, тобто в загальному випадку $s = 0$, $p = 1$.

Другий випадок $A > B$. Нехай $A_{10} = 6_{10} = 0110_2$, а $B_{10} = 5_{10}$. Тоді різниця A і B буде дорівнювати:

$$0110$$
$$+1011$$
$$\overline{1\ 0001}$$

Для отриманого результату $s = 0001$, а перенесення $p = 1$, тобто в загальному випадку $s \neq 0$, $p = 1$.

Третій випадок $A < B$. Нехай $A_{10} = 5_{10}$, а $B_{10} = 6_{10}$. Число B у доповняльному коді дорівнює 1010_2 . Тоді різниця A і B буде дорівнювати:

$$0101$$
$$+1010$$
$$\overline{0\ 1111}$$

Для отриманого результату $s = 1111$, а перенесення $p = 0$, тобто в загальному випадку $s \neq 0$, $p = 0$.

З урахуванням отриманих результатів побудована схема порівняння кодів з використанням повного 4-розрядного суматора, що наведена на рис. 8.22.

8.8. "Небезпечні змагання" сигналів в комбінаційних пристроях

Інерційність реальних логічних елементів приводить до того, що вихідні сигнали з'являються через деякий час, обумовлений тривалістю перехідних процесів. Найважливішим параметром, що характеризує інерційність логічного елемента, є середній час затримки вихідного сигналу стосовно вхідного $t_{3, \text{ср.}}$.

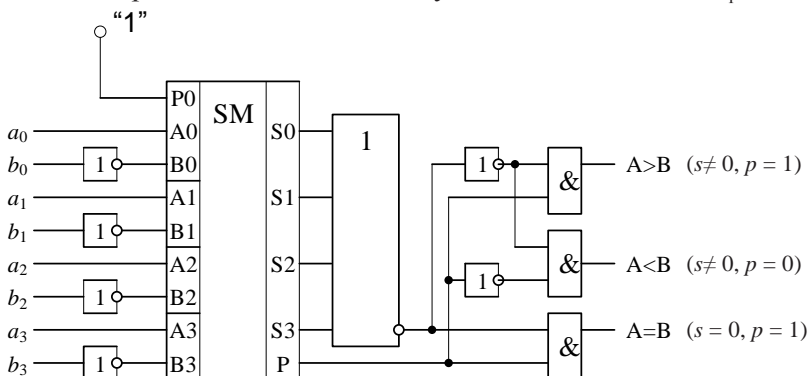


Рис. 8.22. Схема порівняння 4-розрядних кодів з використанням суматора

Логічний елемент можна представити складеним з двох частин – власне логічного елемента (безінерційного елемента, що виконує логічні функції), та елемента затримки D.

У різних частинах комбінаційного пристрою залежно від кількості елементів, що послідовно переключаються під дією вхідного сигналу, перехідний процес буде закінчуватися у різний час.

На виході комбінаційного пристрою можлива поява завад, що порушують роботу пристроїв, які під'єднуються до його виходів. Розглянемо схему комбінаційного пристрою, представлену на рис. 8.23. Характеристичне рівняння такого пристрою

$$y = x_3 x_1 x_4 x_2.$$

Часові діаграми, що ілюструють роботу цієї схеми, наведені на рис. 8.24.

При зміні вхідного сигналу з 1000 на 1111 значення вихідного сигналу має зберегтися рівним 1, однак на виході комбінаційного пристрою за час $3t_{3, \text{ср.}}$ з'являється імпульс завади, обу-

мовленої затримками, що вносять логічні елементи. Якщо така завада порушує роботу включеного на виході пристрою, то маємо справу з “небезпечними станами”.

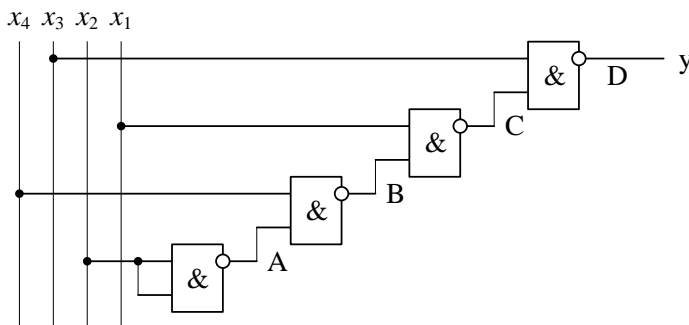


Рис. 8.23. Комбінаційний пристрій з “небезпечними змаганнями” сигналів

Для боротьби з “небезпечними станами” використовують такі заходи.

1. Включають на виході згладжувальний фільтр, наприклад, інтегрувальне коло (використовують рідко).

2. Вводять синхронну передачу сигналів від одного пристрою до іншого за допомогою спеціальних імпульсів синхронізації, що визначають моменти передачі інформації. Пауза між імпульсами синхронізації вибирається такою, щоб під час її дії закінчилися перехідні процеси і на виході пристрою встановилися стаціонарні значення сигналів.

Контрольні питання

1. Що таке функціонально повна система та базис ЛЕ?
2. Синтезуйте логічну схему для реалізації функції

$$F = \overline{x_1} + x_2 + \overline{x_1 + x_2} \text{ на елементах І-НІ.}$$

3. Поясніть призначення та наведіть структурні схеми мультимплексора та демультимплексора.

4. Поясніть призначення та наведіть структурні схеми одноступінчатого, пірамідального та багаступінчатого дешифраторів.

5. Поясніть призначення та наведіть логічну схему шифратора.

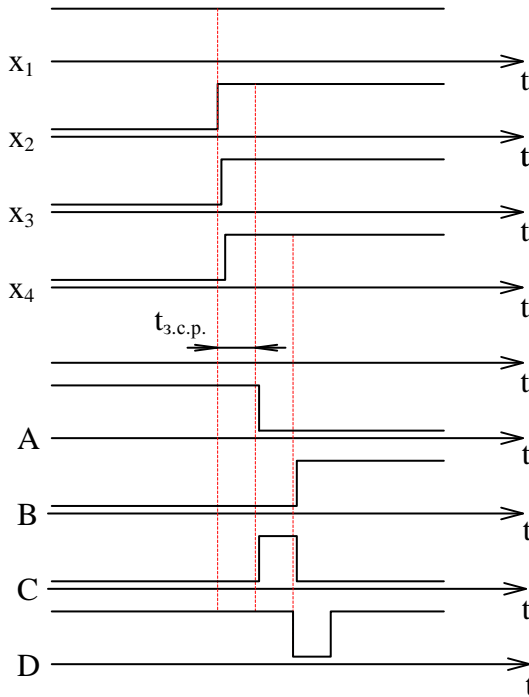


Рис. 8.24. Часові діаграми роботи схеми пристрою з "небезпечними змаганнями" сигналів

6. Запишіть логічну функцію, за допомогою якої реалізується арифметичне додавання однорозрядних двійкових кодів.

7. Чим відрізняються півсуматор від однорозрядного суматора?

8. Поясніть призначення та наведіть структурну схему цифрового компаратора.

9. ТРИГЕРНІ ЕЛЕМЕНТИ

9.1. Призначення та класифікація тригерів

Тригер є пристроєм з двома стійкими станами, що складається із запам'ятовувального елемента (власне тригер) і схеми управління. Схема управління перетворює інформацію, що надходить на її входи, в комбінацію сигналів, що діють на входах власне тригера, стан якого характеризує пристрій в цілому. Можливі й більш прості варіанти тригерів, наприклад такі, в яких взагалі відсутня схема управління.

Власне тригер називають ще бістабільним пристроєм або фіксатором. Здатність запам'ятовувати і зберігати інформацію визначила цілий клас пристроїв.

Тригери розрізняються:

- за функціональною ознакою;
- за способом запису інформації в тригер.

Функціональна ознака визначає тип тригера, тобто його логічне рівняння, що характеризує стан входів і виходів тригера до і після його спрацювання.

Спосіб запису інформації в тригер визначає часову діаграму його роботи. За цією ознакою тригери діляться на дві групи: асинхронні та синхронні. До асинхронних тригерів запис інформації здійснюється безпосередньо з надходженням інформаційного сигналу на його вхід. Запис інформації в синхронні (тактовані) тригери здійснюється тільки за наявності тактового імпульсу на додатковому тактовому вході. Тактовані тригери, в свою чергу, діляться на тригери, які працюють за рівнем тактового імпульсу (тригери із статичним управлінням), при цьому приймання інформації відбувається протягом всієї тривалості імпульсу, і на тригери з динамічним входом, в яких запис інформації проводиться за фронтом або спадом тактового імпульсу. В інший час, незалежно від рівня тактового імпульсу, тригер не сприймає інформаційні сигнали, а отже, не змінює свого стану.

В зарубіжній інженерній практиці всі тригерні схеми ді-

ляться на дві групи.

Перша з них – *flip-flop* – характеризується тим, що вибірка вхідних сигналів і відповідна зміна вихідних станів відбувається в моменти дії тактових сигналів (синхронні тригери).

Особливість другої групи схем – *latch* – полягає в тому, що вони змінюють свій стан при зміні вхідних сигналів незалежно від наявності чи відсутності тактових сигналів.

Тригери відносяться до класу пристроїв, що мають так званий парафазний вихід. Це означає, що стан тригера одночасно визначається прямим та інверсним значеннями на його виходах Q і \bar{Q} (Q і \bar{Q} – внутрішні змінні).

Вихідні значення стану тригера в таблиці перемикань (істинності) прийнято відображати такими символами:

- 0 – ($Q = 0$; $\bar{Q} = 1$);
- 1 – ($Q = 1$; $\bar{Q} = 0$);
- Q – стан тригера не змінюється при зміні інформації на його вході ($Q^n = Q^{n+1}$ або $Q^t = Q^{t+1}$);
- \bar{Q} – стан тригера змінюється на протилежний;
- X – невизначений стан тригера (цим символом позначається також заборонена комбінація значень змінних на вході тригера).

Залежність значень сигналів на виходах тригера від значень сигналів на його входах задається, як правило, у вигляді таблиці переходів тригера. Загальне число комбінацій змінних на вході дорівнює 2^n , де n – число зовнішніх змінних. При кожному наборі зовнішніх змінних тригер може знаходитися в одному з двох стійких станів. Таким чином, загальна кількість рядків у таблиці дорівнює $2 \cdot 2^n$. Повну таблицю переходів тригера, як правило, не використовують, оскільки вона ускладнює сприйняття. Скорочення таблиці виконується за допомогою виключення тих станів, які є очевидними для нормального функціонування тригера і характерними для всіх типів тригерів або тригерів даного класу.

Відомо, що теоретично можна створити 5^{2^n} типів тригерів, де n – число зовнішніх змінних. Проте практичне застосу-

вання знайшла обмежена кількість типів, серед яких найбільш поширеними є, наприклад, RS, T, D, JK та комбіновані RSD і RSJK-тригери.

9.2. Асинхронні та синхронні RS-тригери

9.2.1. Асинхронный RS-тригер

RS-тригером називається логічний пристрій з двома стійкими станами, що має два інформаційних входи R (reset – скидання у початковий стан, за який прийнято "0«») та S (set – встановлення у "1«») При різних комбінаціях сигналів на інформаційних входах буде спостерігатися така ситуація:

- $S = 1, R = 0$ – тригер встановлюється до стану "1«;»- $R = 1, S = 0$ – тригер скидається до стану "0«;»- $R = 0, S = 0$ – тригер зберігає попередній стан $Q^t = Q^{t+1}$;

- $R = 1, S = 1$ – заборонена комбінація.

Зміна станів RS-тригера наведена в табл. 9.1, а умовне графічне позначення на схемах – на рис. 9.1.

Таблиця 9.1. Таблиця переходів RS-тригера

S	R	Q^t	Q^{t+1}
1	0	1	1
		0	1
0	1	1	0
		0	0
0	0	1	1
		0	0
1	1	X	

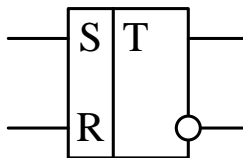


Рис. 9.1. Умовне графічне позначення RS-тригера

Одночасна подача двох активних сигналів на входи S та R заборонена, а якщо така ситуація все ж виникає, то стан тригера вважається невизначеним.

Зміну станів тригера можна показати за допомогою часових діаграм сигналів на його входах та виході (рис. 9.2).

На рис. 9.2 показано, що тригер перемикається з деякою затримкою, величина якої залежить від технології виготовлення

та серії інтегральних мікросхем. Практично, в залежності від серії, затримка може складати від 10 до 300 нс.

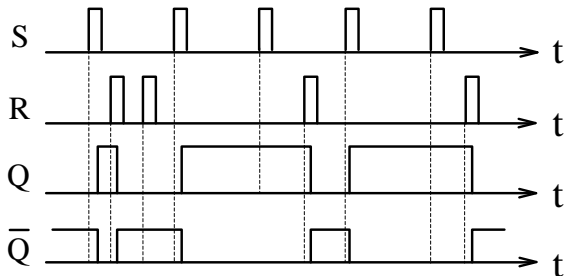


Рис. 9.2. Часові діаграми роботи RS-тригера

Ще однією формою опису функціонування тригера є його характеристичне рівняння:

$$Q^{t+1} = S + \bar{R}Q^t. \quad (9.1)$$

Вибравши як елементну базу базис АБО-НІ, перетворимо (9.1), використовуючи закон заперечення і правило де Моргана, до виду $Q_{t+1} = \overline{R + S + Q_t}$, $\bar{Q}_{t+1} = \overline{S + R + \bar{Q}_t}$.

Схема, що відповідає цьому виразу, наведена на рис. 9.3.

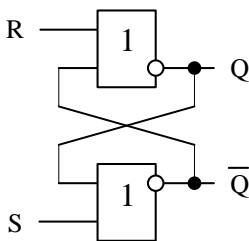


Рис. 9.3. Схема RS-тригера

9.2.2. Асинхронный RS-тригер з інверсними входами

RS-тригером з інверсними входами називається логічний пристрій з двома стійкими станами, що має два інформаційних входи R (reset – скидання у початковий стан, за який прийнято "0") та S (set – встановлення у "1"). При різних комбінаціях сигналів на інформаційних входах буде спостерігатися така ситуація:

- $S = 0, R = 1$ – тригер встановлюється до стану "1";
- $R = 0, S = 1$ – тригер скидається до стану "0";
- $R = 1, S = 1$ – тригер зберігає попередній стан $Q^t = Q^{t+1}$;
- $R = 0, S = 0$ – заборонена комбінація.

Зміна станів RS-тригера наведена в табл. 9.2, а умовне графічне позначення на схемах – на рис. 9.4.

Таблиця 9.2. Таблиця переходів RS-тригера з інверсними входами

S	R	Q^t	Q^{t+1}
0	1	1	1
		0	1
1	0	1	0
		0	0
1	1	1	1
		0	0
0	0	X	

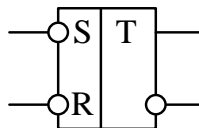


Рис. 9.4. Умовне графічне позначення RS-тригера з інверсними входами

Однчасна подача двох активних сигналів (для цього типу тригерів це рівень логічного "0") на входи S і R заборонена, а якщо така ситуація все ж виникає, то стан тригера вважається невизначеним.

Зміну станів тригера можна показати за допомогою часових діаграм сигналів на його входах та виході (рис. 9.5).

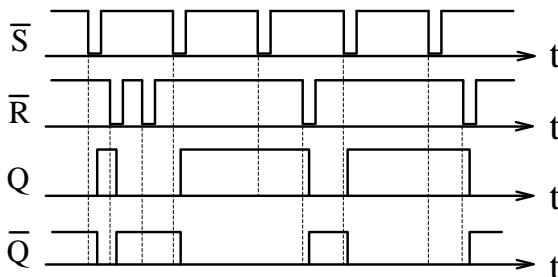


Рис. 9.5. Часові діаграми роботи RS-тригера з інверсними входами

Вибравши як елементну базу базис І–НІ, перетворимо (9.1), використовуючи правило де Моргана $Q_{t+1} = \overline{\overline{S} \overline{R} Q_t}$,

$$\bar{R} + \bar{S} = 1.$$

Схема RS-тригера з інверсними входами, побудована в базисі І-НІ, наведена на рис. 9.6.

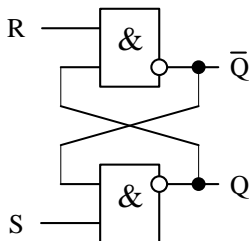


Рис. 9.6. Схема RS-тригера з інверсними входами

9.2.3. Синхронний RS-тригер

На відміну від асинхронних тригерів, що перемикаються при подачі на один з входів активного рівня, синхронний тригер перемикається тільки при наявності дозволяючого (синхронізуючого) сигналу на вході управління, що частіше носить назву входу синхронізації. А сигнали, що подають на цей вхід, називаються синхросигналами або синхроімпульсами (CI).

На рис. 9.7 наведена схема синхронного RS-тригера. Він складається з схеми управління (два логічних елемента І) та самого тригера. Як слідує з аналізу схеми, для того, щоб на входах RS-тригера (на рис. 9.7 позначені як R' та S') з'явився сигнал, необхідно, щоб на вході CI була сформована "1".

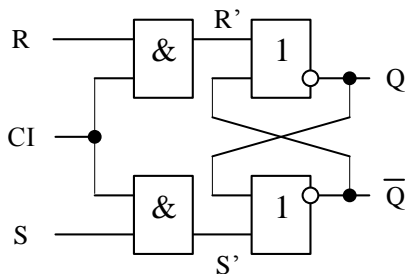


Рис. 9.7. Схема синхронного RS-тригера

При відсутності сигналу на вході CI на виходах логічних елементів І формується рівень логічного "0", а RS-тригер збері-

гає попередній стан. Якщо сигнали на входах S та CI (або на входах R та CI) будуть збігатися у часі, то тоді, відповідно, на виході верхнього (нижнього) логічних елементів І буде формуватися сигнал управління і RS-тригер буде перемикатися.

Часові діаграми, що пояснюють роботу синхронного RS-тригера, наведені на рис. 9.8.

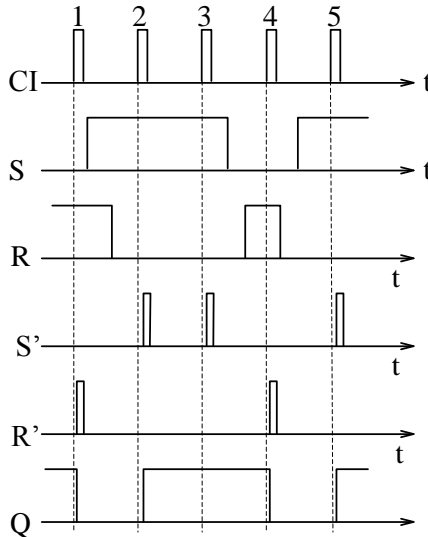


Рис. 9.8. Часові діаграми роботи синхронного RS-тригера

9.3. Тригери Т-типу

Тригер Т-типу є логічним пристроєм з двома стійкими станами й одним інформаційним входом, що змінює стан на виході щоразу, коли на його вхід надходить імпульс. Цей тригер ще називається лічильним тригером. Логічне рівняння такого тригера має вигляд:

$$Q^{t+1} = \bar{T}^t Q^t + T^t \bar{Q}^t.$$

Даний тригер залежно від позначення може спрацьовувати або за фронтом імпульсу (/), або за його спадом (\). Функціональне позначення Т-тригера, що спрацьовує за фронтом, наведено на рис. 9.9, а, а тригера, що спрацьовує за спадом вхі-

дного імпульсу – на рис. 9.9, б).

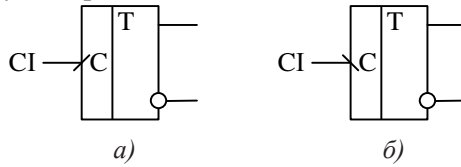


Рис. 9.9. Умовне графічне позначення T-тригера

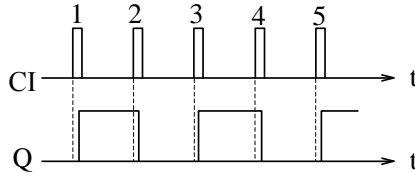


Рис. 9.10. Часові діаграми роботи T-тригера

При виконанні тригера в базисі АБО-НІ і за умови, що тригер керується перепадами з 1 в 0 на вході С (\downarrow), отримаємо схему, показану на рис. 9.11.

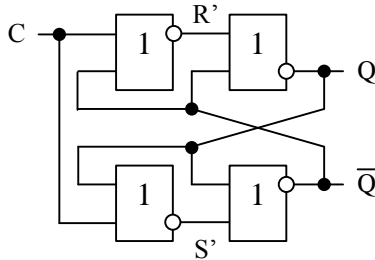


Рис. 9.11. Схема T-тригера з імпульсним управлінням в базисі АБО-НІ

9.4. Тригери D-типу

Тригер D-типу є логічним пристроєм з двома стійкими станами й одним інформаційним входом D, що запам'ятовує інформацію, яка надходить на вхід D в момент приходу синхроімпульсу на вхід C (синхронізація). Логічне рівняння такого тригера має вигляд

$$Q^{t+1} = D'.$$

Значення вихідного сигналу в момент часу $t+1$ збігається з кодом вхідного сигналу в момент часу t . Умовне позначення

D-тригера наведено на рис. 9.12, а варіант реалізації на елементах АБО-НІ – на рис. 9.13.

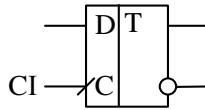


Рис. 9.12. Умовне графічне позначення D-тригера

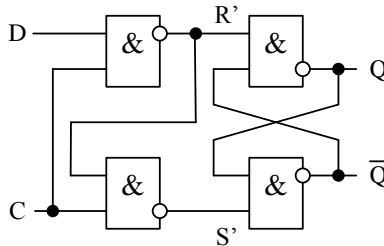


Рис. 9.13. Варіант реалізації D-тригера на елементах І-НІ

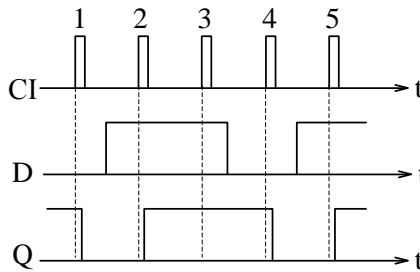


Рис. 9.14. Часові діаграми роботи D-тригера

Якщо вхід D з'єднати з інверсним виходом тригера, то $D = \bar{Q}$ і значення вихідного сигналу в момент часу $t+1$ збігається з кодом на інверсному виході тригера в момент часу t , тобто $Q^{t+1} = \bar{T}^t Q^t + T^t \bar{Q}^t$ і тригер працює в режимі лічильного тригера. Схема включення D-тригера в режим T-тригера наведена на рис. 9.15.

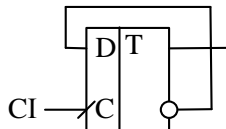


Рис. 9.15. Схема включення D-тригера в режим T-тригера

Слід зауважити, що Т-тригери у вигляді окремих інтегральних мікросхем не випускаються, а реалізуються на D-тригерах або JK-тригерах.

9.5. Тригери JK-типу

Тригер JK-типу є логічним пристроєм з двома стійкими станами і двома інформаційними входами J і K, що за умови $J=K=1$ здійснює інверсію попереднього стану (працює в режимі лічильного тригера), а в інших випадках працює як RS-тригер. При цьому вхід J еквівалентний входу S, а вхід K – входу R. Логічне рівняння, що описує роботу тригера, має вигляд

$$Q^{n+1} = \overline{K}^n Q^n + J^n \overline{Q}^n.$$

Слід зазначити, що JK-тригер відноситься до синхронних тригерів і тому всі перемикання цього тригера відбуваються тільки в моменти надходження синхроімпульсів на його вхід синхронізації С. Умовне графічне позначення JK-тригера наведено на рис. 9.16.

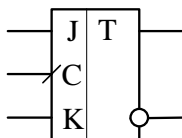


Рис. 9.16. Умовне графічне позначення JK-тригера

В табл. 9.3 наведено переходи JK-тригера при різних значеннях сигналів на входах J та K.

Табл. 9.3. Таблиця переходів JK-тригера

J ^t	K ^t	Q ^t	Q ^{t+1}
0	0	0	0
		1	1
1	0	0	1
		1	1
0	1	0	0
		1	0
1	1	0	1

		1	0
--	--	---	---

JK-тригер найбільш універсальний з розглянутих тригерів. Він може працювати в режимі Т- або D-тригера, що витікає з таблиці функціонування. На рис. 9.17 наведено схеми підключення JK-тригера для реалізації цих видів тригерів.

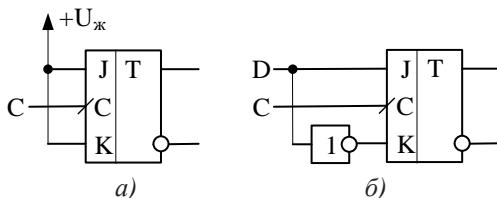


Рис. 9.17. Схеми підключення JK-тригера для реалізації Т-тригера (а) та D-тригера (б)

9.6. Універсальні RSD- та RSJK-тригери

Універсальні тригери – тригери, що можуть працювати в різних режимах і як різні типи тригерів. Умовні графічні позначення універсальних тригерів наведені на рис. 9.18.

Розглянемо роботу таких тригерів.

RSJK-тригер (рис. 9.18, а) – це логічний пристрій з двома стійкими станами і чотирма інформаційними входами RSJK, що за умови $S = R = 0$ працює як JK-тригер, а у всіх інших випадках як асинхронний RS-тригер. Тобто, якщо:

- $S = 1, R = 0$, то тригер перемикається до "1";
- $S = 0, R = 1$, то тригер перемикається до "0";
- $S = R = 1$ – заборонена комбінація.

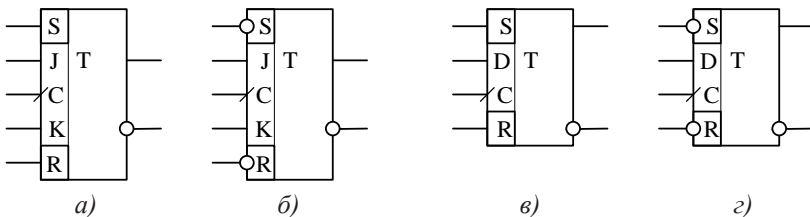


Рис. 9.18. Умовне графічне позначення RSJK- та RSD-тригерів з прямими та інверсними входами

RSJK-тригер з інверсними входами (рис. 9.16, б) – це логічний пристрій з двома стійкими станами і чотирма інформаційними входами RSJK, що за умови $S = R = 1$ працює як JK-тригер, а у всіх інших випадках – як асинхронний RS-тригер. Тобто, якщо:

- $S = 0, R = 1$, то тригер перемикається до "1";
- $S = 1, R = 0$, то тригер перемикається до "0";
- $S = R = 0$ – заборонена комбінація.

Аналогічно функціонують і RSD-тригери. Для тригера з прямими входами R та S активний рівень на цих входах "1«»заборонена комбінація $S = R = 1$, а для RSD-тригера з інверсними входами S та R активний рівень на цих входах "0", а заборонена комбінація $S = R = 0$.

На рис. 9.19 наведений приклад реалізації RSD-тригера з інверсними входами в базисі І-НІ. Зі схеми витікає, що входи R та S мають пріоритет, оскільки вони зумовлюють, чи будуть працювати логічні елементи, до яких вони під'є'нані.

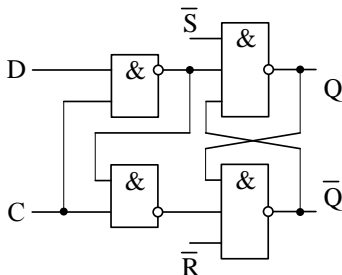


Рис. 9.19. Приклад реалізації RSD-тригера з інверсними входами в базисі І-НІ

Універсальні D- і JK-тригери широко використовуються при побудові лічильників, регістрів, суматорів, пристроїв управління, розподільників імпульсів, різних цифрових автоматів, при синтезі довільних тригерних структур та ін.

9.7. Приклади використання тригерів

Стартостопний пристрій у пристроях управління – сукупність деякого з RS-тригерів і керованого їм вентиля. На рис.

9.20, а, б показані найпростіша реалізація стартостопного пристрою на трьох двохходових вентилях і часова діаграма його роботи. Зазначимо, що конкретна реалізація розглянутих схем передбачається виконаною на елементах ТТЛ-серій, наприклад К555 або КР1533. Тривалість імпульсів управління τ_1 і τ_2 (стартостопних імпульсів) має бути достатньої для спрацьовування RS-тригера й одночасно менше інтервалу часу між стартостопними імпульсами (для виключення можливості появи на входах забороненої комбінації).

Якщо стартостопне управління формується за допомогою контактних перемикачів типу реле, кнопок та ін., то для поліпшення динамічних властивостей тригера і підвищення його завадостійкості "вільні" виводи мікросхем мають бути зафіксовані на рівнях "0" та "1". Прикладом може бути схема, зображена на рис. 9.20, в. В цій схемі опори резисторів R_1 та R_2 вибираються зі співвідношення $R_1 = R_2 \leq U_{\text{вх}}^0 / I_{\text{вх}}^0$.

При використанні даної схеми потрібно враховувати таку обставину: якщо сигнал "Старт" є одночасно сигналом установки нуля (фрагмент схеми, що виконаний пунктиром), то схема неприйнятна. Після короткочасного замикання контактів "Старт" в колі установки нуля має бути рівень "1", а в даній схемі $U_{\text{вих}}^0 + R_1 I_{\text{вх}}^0$, де $U_{\text{вих}}^0$ – напруга на виході, $R_1 I_{\text{вх}}^0$ – спад напруги на резисторі від струму верхнього вентиля RS-тригера. Ця напруга сприймається в колі установки нуля як рівень "0", що блокує роботу елементів, для яких призначено коло установки нуля.

Зазначений недолік не можна усунути збільшенням опорів резисторів R_1 та R_2 , тому що схема буде практично не захищеною від завад. Як правило, ця схема використовується як генератор одиночних імпульсів (рис. 9.20, з), що усуває деренчання контактів перемикача.

До переваг схем, зображених на рис. 9.20, в, з, можна віднести те, що резистори R_1 та R_2 не споживають потужності від джерела при розімкнутих контактах (точніше, ця потужність мізерно мала).

Недолік, зазначений для схеми на рис. 9.20, в, усунутий у

схемі, показаній на рис. 9.20, *д*, однак якщо за цією схемою виконувати генератор одиночних імпульсів, то резистор, у нормальному стані замкнутий на землю, буде споживати значну потужність від джерела живлення. Можлива реалізація RS-тригера на одному вентилі І та одному вентилі АБО (рис. 9.20, *е*). У даного тригера на відміну від нормальних схем виходи не додаткові і управління на входах здійснюється перепадами різної полярності.

Тригер може бути особливо корисним у випадках, коли необхідно виключити можливість змагань. Як впливає з діаграми рис. 9.20, *ж*, на якій враховані затримки поширення сигналів, сигнал на виході y_1 цілком лежить усередині інтервалу часу, що відповідає тривалості сигналу y_2 . Неважко переконатися, що на парах елементів І-НІ, АБО-НІ; АБО, І-НІ не можна реалізувати тригерну структуру з використанням двох перехресних кіл зв'язу.

Запропонована реалізація RS-тригера на одному вентилі І або одному вентилі АБО (рис. 9.20, *з*, *і*). Припустимо, що управління схемою здійснюється контактами, які замикаються. На вході x_1 забезпечується рівень "1" через резистор R від джерела живлення. Якщо тепер короткочасно замкнути контакт "Старт", то на виході встановиться рівень "1" і буде утримуватися, тому що $x_2 = y$. Короткочасне замикання контакту "Стоп" забезпечує появу "0" на виході й утримання його, при цьому по входу $x_2 = y$. Робота схеми, зображеної на рис. 9.20, *і* відбувається аналогічно.

Схеми рис. 9.20, *з*, *і* мають незвичайну властивість – між сигналом по одному з входів і сигналом на виході тут немає вентильної затримки; однак у тригера є недолік – сигнал на одному з входів обов'язково збігається з сигналом на виході. При реалізації конкретної схеми функціонування її має забезпечуватися з передбаченням заходів захисту вентилів при одночасній дії сигналів "Старт" і "Стоп". В усіх схемах, показаних на рис. 9.20, управління RS-тригерами здійснюється імпульсними сигналами для усунення заборонених комбінацій на їх входах. Однак іноді потрібно забезпечити комбіноване управління: по одному входу – імпульсне, а по іншому – перепадом потенціалів. Приклади

схем, що задовольняють цим вимогам, наведені на рис. 9.21.

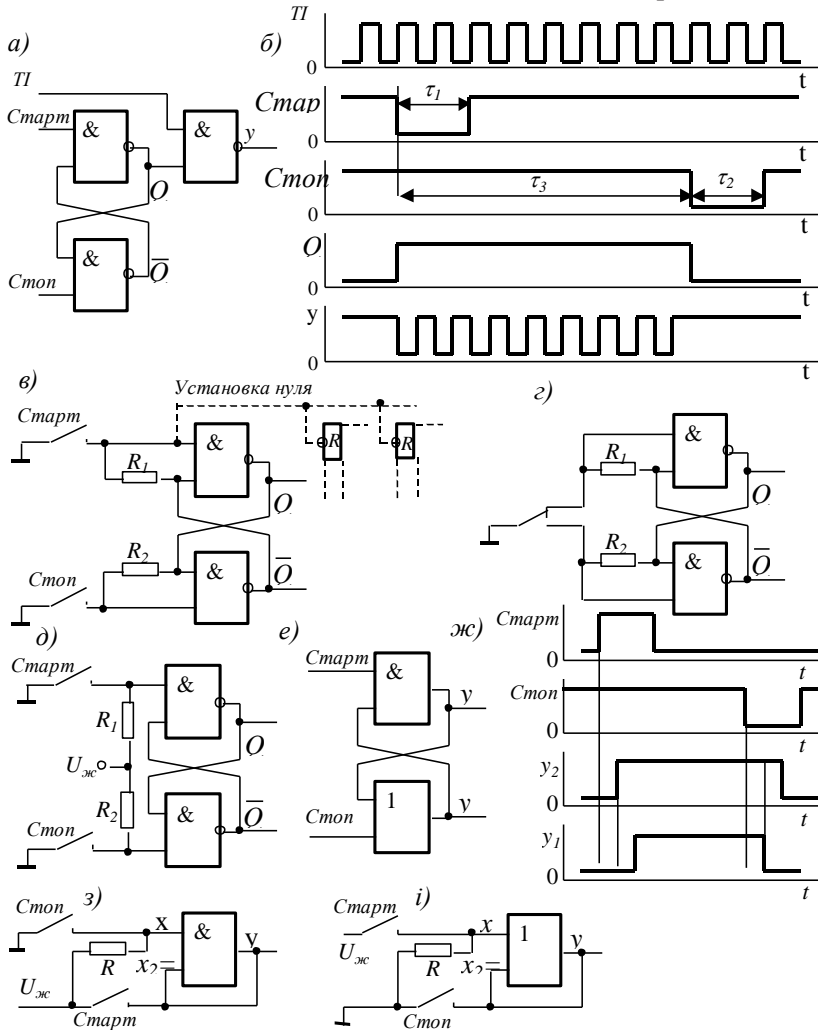


Рис. 9.20. Варіанти схем стартозупинних пристроїв з імпульсним управлінням: а – базова схема; б...е, з, і – реалізація структури RS-триггера; б, ж – відповідні часові діаграми роботи схем а та е

Розглянемо роботу схеми рис. 9.21, а. Припустимо, що тригер перебуває в стані "0", а на входах R і S встановлені рівні

логічної "1" (рис. 9.21, б). Тому що на вході D постійно присутній рівень "0", переходи 0-1 підтверджують нульовий стан. Тригер можна встановити в стан "1" тільки імпульсним сигналом "Старт", а в стан "0" тригер перейде з приходом першого переходу 0-1 на вході "Стоп". Робота інших схем відбувається аналогічно.

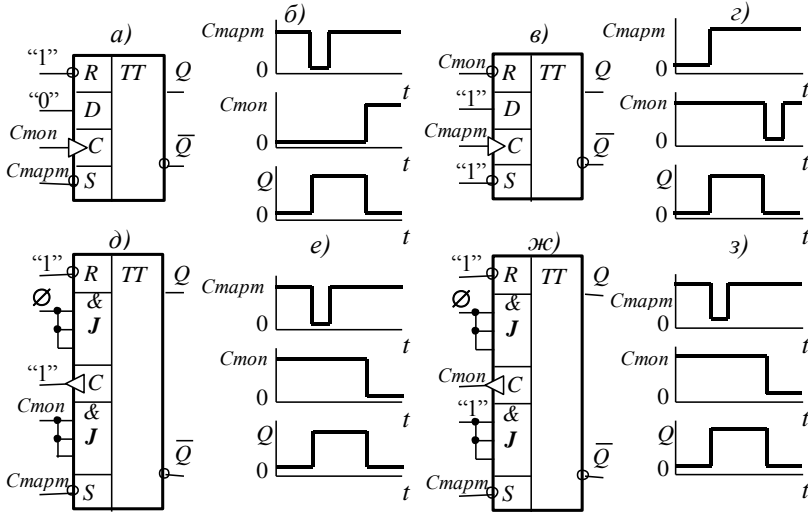


Рис. 9.21. Варіанти схем стартостопних пристроїв з комбінованим управлінням: а, в – відповідно на D- тригері при $D=0$ та $D=1$; б, г – відповідні часові діаграми роботи схем а та в; д, ж – відповідно схеми на JK- тригері в асинхронному та тактовому режимах; е, з – відповідні часові діаграми роботи схем д та ж

На рис. 9.22 наведені варіанти стартостопних пристроїв з управлінням переходами по обох входах.

В цифрових пристроях різного призначення часто виникає задача виділення переходів 1-0 та 0-1 асинхронних сигналів, що з'являються в довільний момент часу, з одночасною прив'язкою виділених переходів до моменту часу, обумовленого тактовими імпульсами.

Приклади таких схем наведені на рис. 9.23. Розглянемо роботу схеми, показаної на рис. 9.23, а. Задача, що вирішує дана схема, може бути сформульована у такий спосіб: розробити цифровий пристрій, який виділяє перехід 1-0 асинхронного сигнала.

лу x та поміщає виділений перехід у момент часу, який відповідає першому переходу 0-1 тактових імпульсів і виникає відразу після появи переходу 1-0 сигналу x , при цьому тривалість вихідного сигналу цифрового пристрою має дорівнювати тривалості тактового імпульсу.

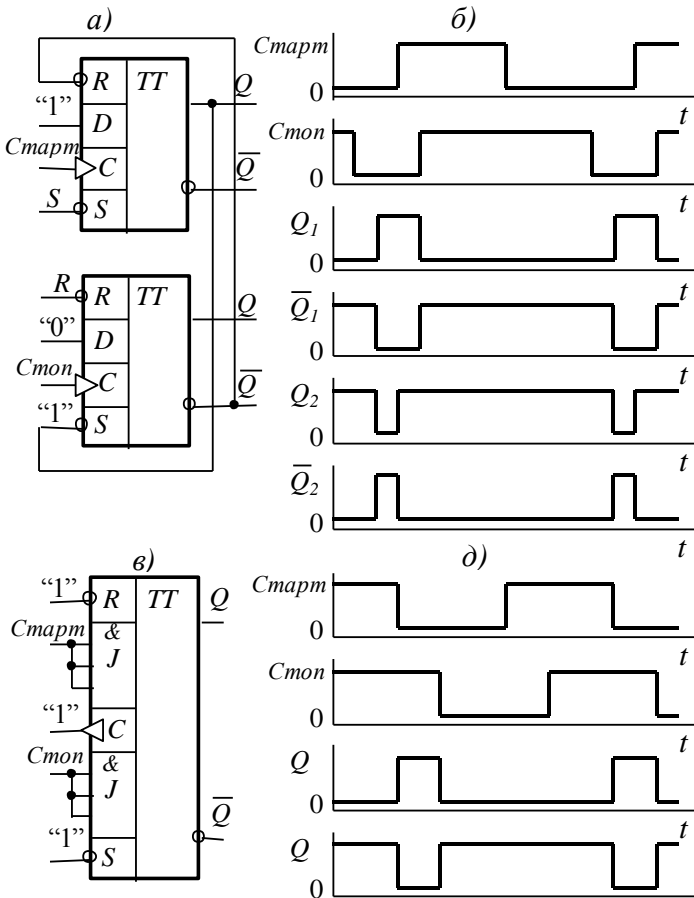


Рис. 9.22. Варіанти схем стартостопних пристроїв з управлінням переходами: а – схема на D-триггерах; в – схема на JK-триггері; б, г – відповідні часові діаграми роботи схем а та в

Якщо подати сигнал x на вхід D першого D-триггера, то кожен перехід 0-1 сигналу TI переводить тригер до стану $Q_1=x$.

Якщо при цьому вихід Q_1 з'єднати з входом D другого D-тригера, то кожен перехід 0-1 сигналу буде переводити тригер у стан $Q_2 = Q_1 = x$ при зміщенні зі зсувом на половину періоду тактових імпульсів (при шпаруватості, що дорівнює 2). З часової діаграми (рис. 9.23, б) випливає, що вихідний сигнал у описується рівнянням $y = \overline{Q_1}Q_2$.

Зазначимо, що отриманий пристрій ніяк не реагує на перехід 0-1 сигналу x .

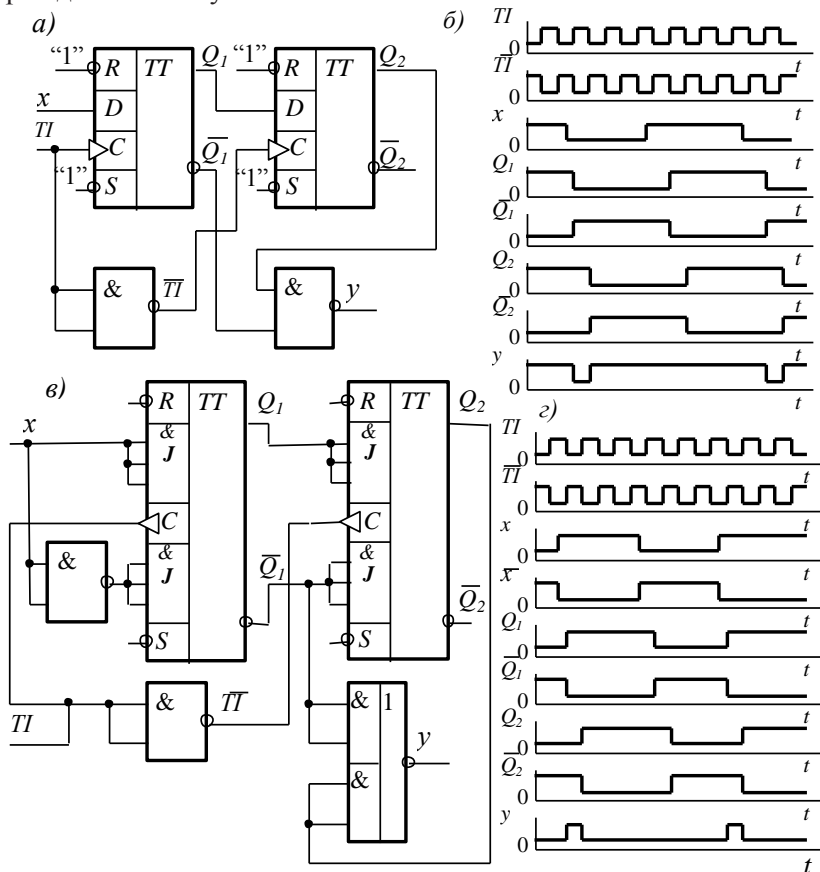


Рис. 9.23. Схеми пристроїв виділення і стробіювання переходів 1-0 та 0-1: а – 1-0 на D-тригерах; в – 0-1 на JK-тригерах; б, г – відповідні часові діаграми роботи схем а та в

Робота схеми, зображеної на рис. 9.23, в, що виділяє перехід 0-1 сигналу x та виконана на універсальних JK-тригерах, пояснюється аналогічно.

Контрольні питання

1. Що таке тригерні пристрої (тригери)? Перерахуйте області їхнього використання.
2. Наведіть класифікацію тригерів за визначальними ознаками.
3. Якими параметрами і характеристиками визначається робота тригера в статичному та динамічному режимах?
4. Якими способами можна описати закон функціонування тригера?
5. Перелічіть різновиди RS-тригера.
6. Чим принципово відрізняється робота схеми Т-тригера від роботи RS-тригера?
7. Що таке лічильний тригер? Що таке лічильний режим роботи універсального тригера?
8. Що таке синхронні тригери?
9. Сформулюйте визначення універсального тригера.
10. Опишіть роботу універсального JK-тригера в асинхронному і синхронному режимах.

10. РЕГІСТРИ

10.1. Загальні зауваження

Регістрами називаються пристрої, що виконують функції прийому, збереження і передачі інформації. Інформація в регістрі зберігається у вигляді числа (слова), представленого комбінацією нулів та одиниць.

Регістри виконуються, як правило, на основі RS-, D- або JK-тригерів.

За допомогою регістрів можуть здійснюватися операції перетворення кодів (наприклад, з послідовного в паралельний і навпаки), а також деякі логічні операції (порозрядне логічне додавання й ін.). Найбільше застосування регістри знаходять для тимчасового збереження інформації.

Основною ознакою, за якою розрізняють регістри, є спосіб запису інформації в регістр. За цією ознакою регістри можна розділити на 3 групи:

- паралельні – інформація записується одночасно в усі тригери (розряди) регістра;
- послідовні (зсуву) – інформація записується спочатку в перший тригер, потім передається в другий і т.д.;
- універсальні – працюють в будь-якому з режимів.

Послідовні регістри діляться на регістри зсуву та реверсивні. В звичайних регістрах зсуву інформація зсувається в один бік, а в реверсивних – в обидва.

В паралельних регістрах запис числа здійснюється в усі розряди регістра одночасно (паралельним кодом).

В послідовних регістрах запис коду здійснюється починаючи з молодшого (або старшого) розряду шляхом послідовного зсуву коду тактовими імпульсами. Регістри паралельно-послідовного типу мають входи як для паралельного, так і для послідовного запису інформації.

10.2. Паралельні регістри на D-тригерах

В паралельному регістрі інформація одночасно записується по всіх входах. Прикладом такого регістра є паралельний регістр на тригерах D-типу (рис. 10.1).

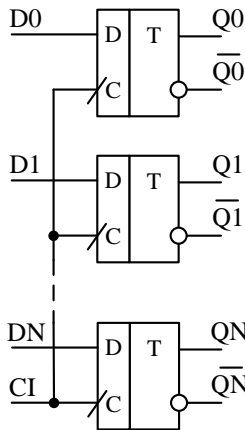


Рис. 10.1. Схема параллельного регистра на триггерах D-типу

В паралельному регістрі в момент надходження синхронізатору інформація одночасно записується зі всіх входів D та зберігається в регістрі до моменту приходу наступного синхронізатору. На рис. 10.2 показані часові діаграми роботи паралельного регістра.

Для реалізації паралельних регістрів можна використувати RS-тригери, але для цього потрібно або попередньо встановити тригери регістра в нульовий стан, або порозрядно записувати інформацію. Це ускладнює схему регістра та вимагає додаткових сигналів для управління роботою регістра.

В таблиці 10.1 наведена таблиця переходів одного розряду регістра на RS-тригерах, яку можна отримати при використанні рівняння $Q_i^{t+1} = Q_i^t T_R + X_{2i} T$.

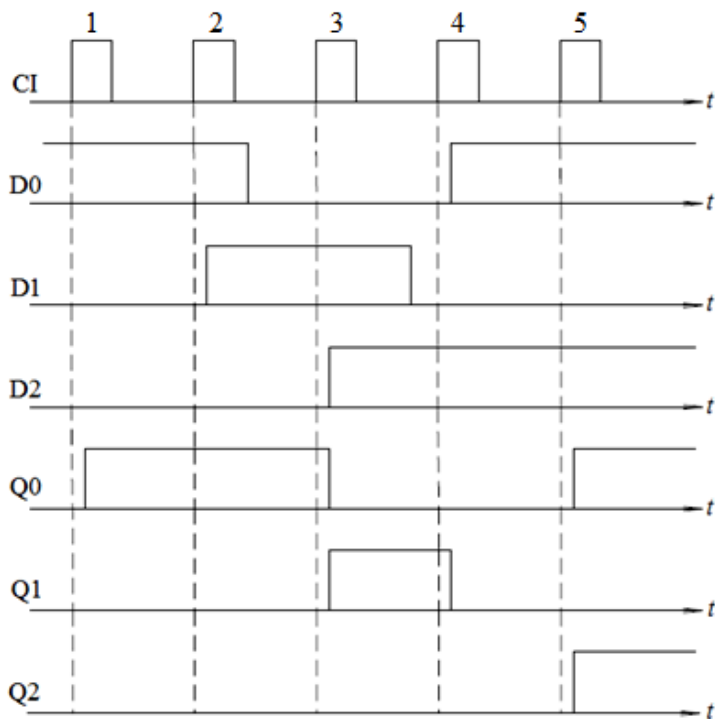


Рис. 10.2. Часові діаграми роботи паралельного регістра

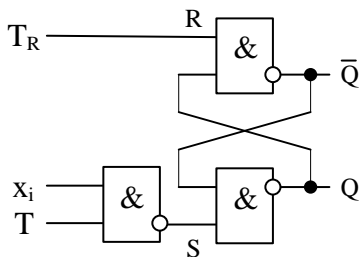


Рис. 10.3. Схема одного розряду паралельного регістра на RS-тригерах

10.3. Послідовні регістри

Послідовні регістри (регістри зсуву), як і паралельні,

призначені для короткочасного збереження інформації, представлені у двійковому коді, будуються на тригерах різних типів. У послідовних регістрах здійснюється логічна операція зсуву коду числа.

Таблиця 10.1. Таблиця переходів одного розряду регістра на RS-тригерах

T_R	T	Q_i^{t+1}	Операція
0	0	0	установка в "0"
0	1	0	установка в "0"
1	0	Q_i^t	збереження інформації
1	1	X_{2i}	запис інформації

За методом введення і виведення інформації послідовні регістри підрозділяються на власне послідовні, паралельно-послідовні та послідовно-паралельні.

В послідовних регістрах інформація вводиться і виводиться в послідовному коді, розряд за розрядом. У паралельно-послідовних регістрах інформація вводиться в паралельному, а виводиться в послідовному коді. У послідовно-паралельному, навпаки, введення відбувається в послідовному коді, а виведення – в паралельному.

Зсув коду може здійснюватися за однократною або багатократною схемами. В однократному регістрі кожен імпульс синхронізації викликає зсув коду на один розряд.

Найбільш зручно вважається побудова регістрів на тригерах D-типу. На рис. 10.4 наведена схема трирозрядного регістра зсуву. На схемі позначення DI відповідає англ. *data input* (вхідні дані). На рис. 10.5 наведені часові діаграми роботи послідовного регістра, а в табл. 10.2 показано, як змінюється стан тригерів регістра в процесі роботи.

Тому що дані зсуваються з часом (кожен тригер зсуває дані на один такт), то такий регістр крім послідовного ще називається регістром зсуву.

Реверсивний регістр зсуву дозволяє зсувати інформацію праворуч чи ліворуч залежно від значення сигналу керування на додатковому вході. На рис. 10.6 показана схема реверсивного

регістра зсуву з використанням цифрових перемикачів. В даній схемі сигнал на вході V керує перемиканням напрямку зсуву інформації в регістрі.

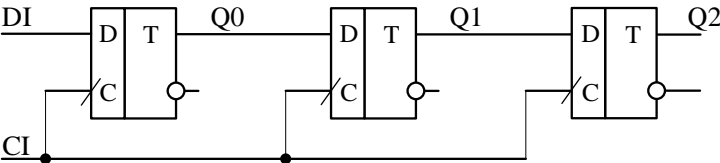


Рис. 10.4. Схема трирозрядного послідовного регістра на D-тригерах

Таблиця 10.2. Таблиця переходів трирозрядного регістра зсуву

№ такту	DI	Q0	Q1	Q2
1	0	0	0	0
2	1	1	0	0
3	0	0	1	0
4	0	0	0	1
5	1	1	0	0

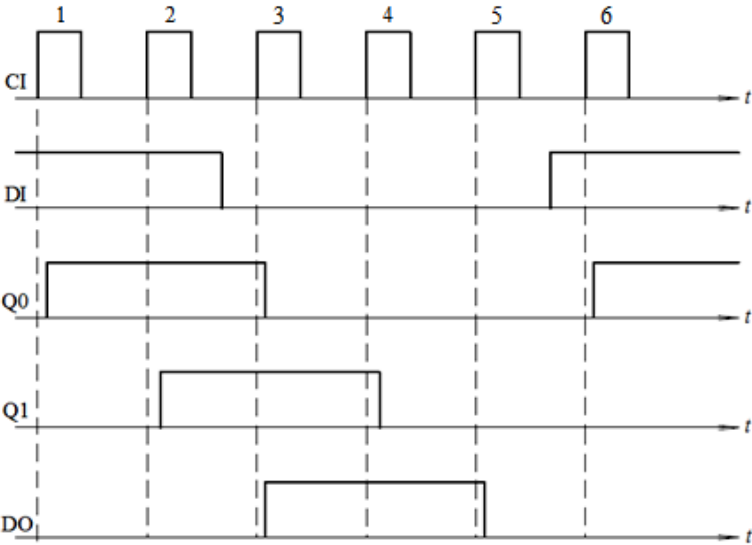


Рис. 10.5. Часові діаграми роботи послідовного регістра

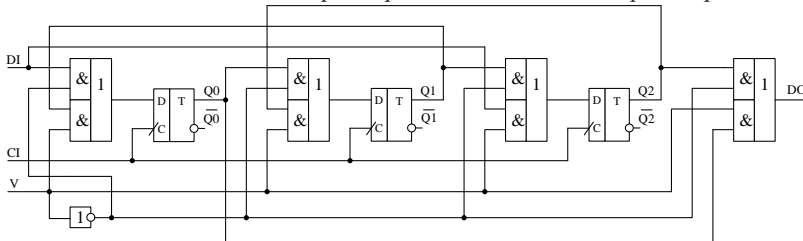


Рис. 10.6. Схема трирозрядного реверсивного регістра зсуву

При подачі на вхід V рівня лог. "0" інформація зсувається праворуч, оскільки сигнал, що формується за допомогою інвертора з сигналу V, дозволяє проходження сигналів через верхній логічний елемент I кожного з електронних перемикачів, тобто праворуч. При подачі на вхід V сигналу рівня лог. "1" дозволяється проходження сигналів через нижній елемент I, що забезпечує передавання інформації у зворотному напрямку.

10.4. Паралельно-послідовні та послідовно-паралельні регістри

Паралельно-послідовні та послідовно-паралельні регістри відрізняються від послідовних наявністю схем паралельного введення розрядів або виведення усіх розрядів регістра.

Найбільш просто реалізується послідовно-паралельний регістр. Достатньо вивести виходи кожного з тригерів на вихід, при цьому послідовний регістр може реалізовувати функції послідовно-паралельного регістра (рис. 10.7).

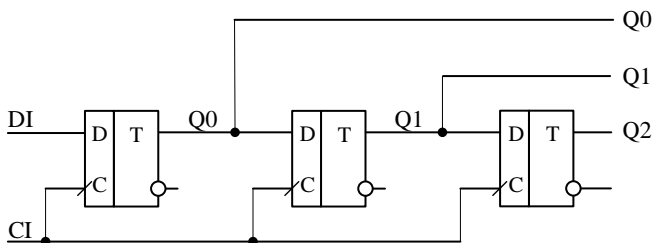


Рис. 10.7. Схема трирозрядного послідовно-паралельного регістра

Для реалізації паралельно-послідовного регістра необхідно організувати запис інформації у кожен з тригерів регістра. На рис. 10.8 наведено один з можливих варіантів організації такого регістра на RSD-тригерах.

Входи R та S використовують для запису паралельного коду. Вхід D використовується для організації функції зсуву.

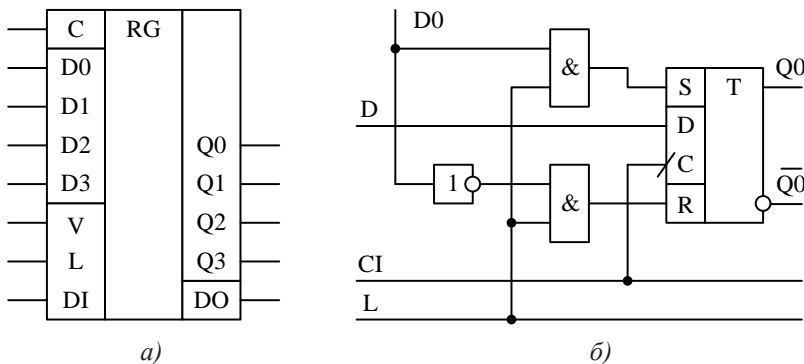


Рис. 10.8. Умовне графічне зображення паралельно-послідовного регістра (а) та схема організації одного розряду (б)

На рис. 10.8, а введено такі позначення: С – вхід синхронізації; D0, D1, D2, D3 – входи паралельного коду; V – вхід керування напрямом зсуву; L – вхід дозволу запису паралельного коду; DI – вхід послідовного коду; DO – вихід послідовного коду; Q0, Q1, Q2, Q3 – виходи паралельного коду.

10.5. Послідовні регістри зі зворотними зв'язками

При введенні в послідовний регістр зворотного зв'язку він перетворюється в замкнуте кільце, у якому при наявності імпульсів зсуву циркулює введена в регістр інформація. Такі регістри називаються кільцевими лічильниками.

На рис. 10.9 наведений приклад побудови трирозрядного регістра, в якому вихідний сигнал подається знову на вхід регістра. Для нормального функціонування такого регістра необхідно в один або два розряди на початку роботи записати "1". На рис. 10.10 наведені часові діаграми, що пояснюють роботу такого

пристрою, який має назву кільцевого лічильника. На початку роботи всі тригери встановлюються в "0" сигналом T_R , а перед подачею синхроімпульсів в перший тригер записується "1" шляхом подачі імпульсу на вхід S.

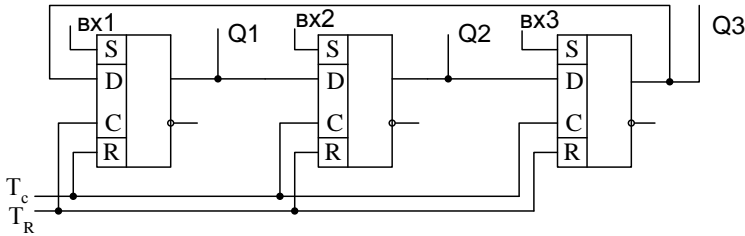


Рис. 10.9. Схема трирозрядного кільцевого лічильника

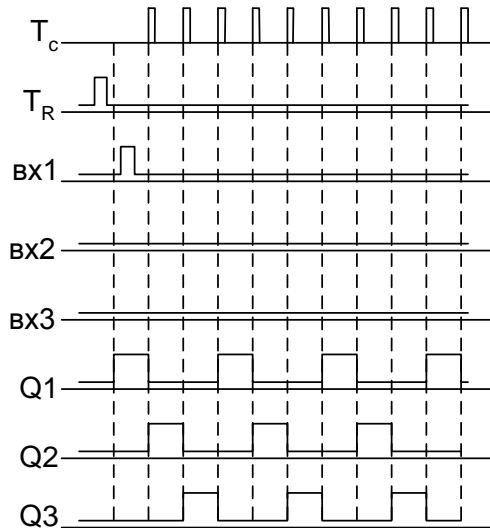


Рис. 10.10. Часові діаграми роботи кільцевого лічильника

Окрім звичайних регістрів випускаються регістри з трьома станами на виході. Такі регістри мають додатковий вхід для перемикання виходу регістра до високоімпедансного стану (наприклад, КР1533ІР23, що є 8-ми розрядним паралельним регістром з вихідними буферними підсилювачами. Перемикання до третього стану здійснюється подачею на вхід \overline{EZ} високого рів-

ня. Перехід до робочого стану здійснюється подачею низького рівня на цей вхід. На рис. 10.11 наведено його умовне графічне позначення та електрична схема.

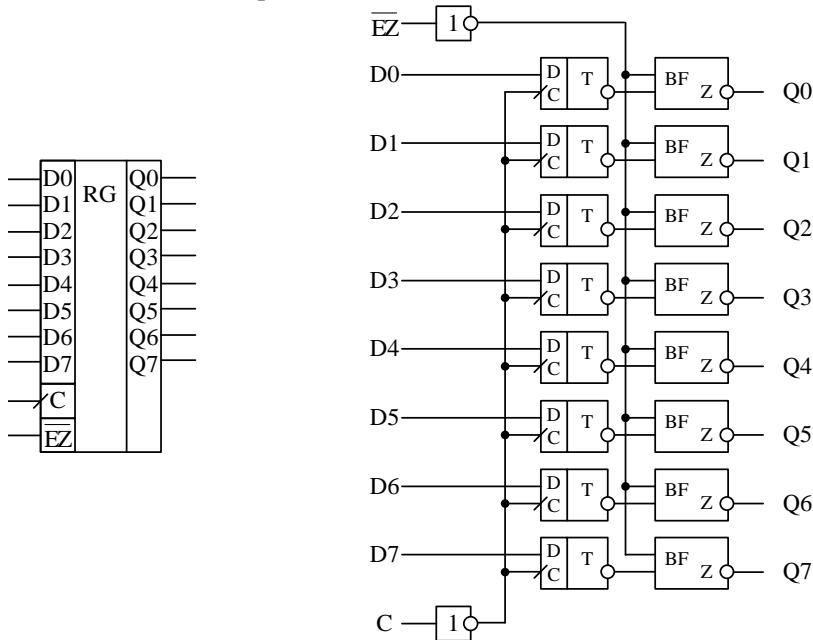


Рис. 10.11. Умовне графічне позначення та схема регістра KP1533IP23

10.6. Приклади використання регістрів

10.6.1. Робота декількох регістрів на спільну шину

При використанні регістрів з трьома станами на виході значно спрощується організація зв'язку декількох джерел інформації з одним приймачем цієї інформації. На рис. 10.12 наведена схема з'єднання двох регістрів з трьома станами з однією шиною.

При подачі рівня логічного "0" на вхід T_2 активується верхній регістр, а виходи нижнього регістра переводяться у високоімпедансний стан. Якщо подати на вхід T_2 рівень логічної "1", то активується нижній регістр і інформація з його виходів

надходить на вихідну шину.

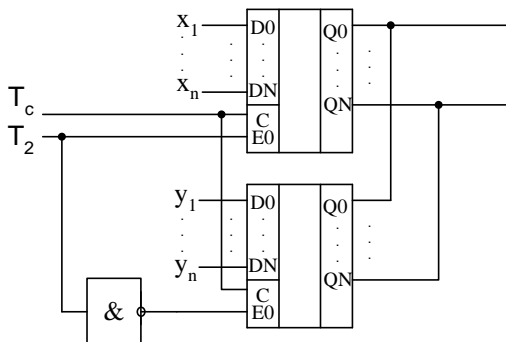


Рис. 10.12. Робота декількох регістрів на спільну шину

10.6.2. Формувач квазігармонічного сигналу

Відомо, що сигнал будь-якої форми можна скласти з елементарних сходинок, частота слідування яких має перевищувати частоту вихідного сигналу у декілька разів. Амплітуда сходинок змінюється з часом, а їх кількість зумовлює точність відтворення форми потрібного сигналу.

На рис. 10.13 наведено схему найпростішого формувача сигналу, що складається з 8 сходинок. Часові діаграми, що пояснюють його роботу, наведені на рис. 10.14. Амплітуда сходинок залежить від значень опорів резисторів $R1 \dots R4$.

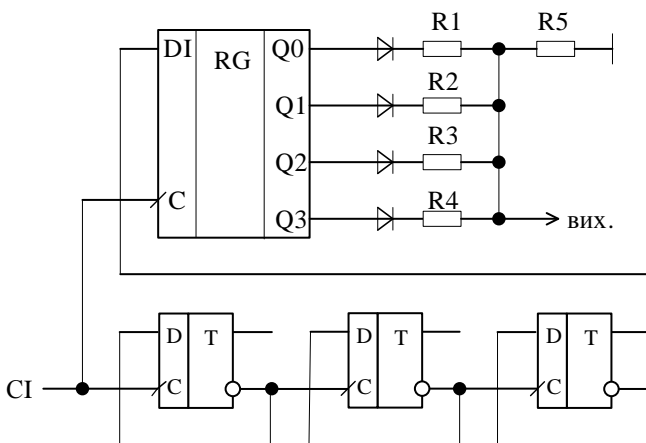


Рис. 10.13. Схема формувача квазігармонічного сигналу

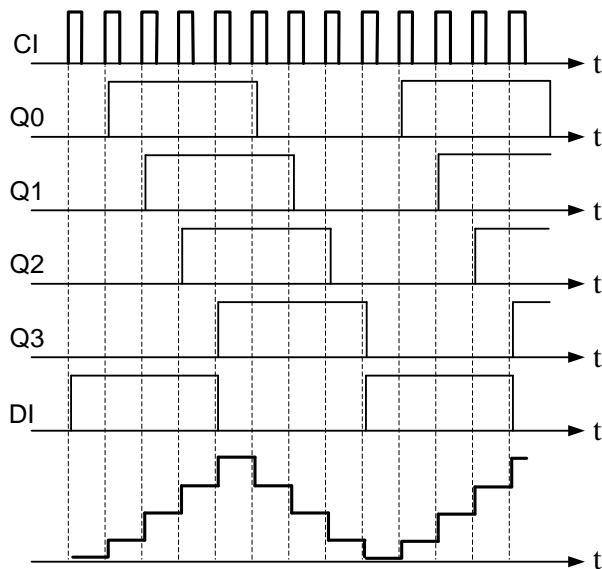


Рис. 10.14. Часові діаграми роботи формувача квазігармонічного сигналу

10.6.3. Перетворювач паралельного коду в послідовний

На рис. 10.15 наведена функціональна схема перетворювача паралельного коду в послідовний.

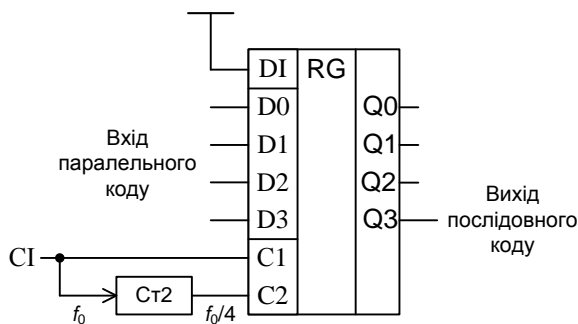


Рис. 10.15. Схема перетворювача паралельного коду в послідовний

На входи $D0 \dots D3$ подається паралельний код, а з виходу $Q3$ знімається послідовний код. Часові діаграми, що пояснюють роботу пристрою, наведені на рис. 10.16. На схемі вхід $C1$ регістра призначений для синхронізації зсуву послідовного коду, а вхід $C2$ – для синхронізації запису паралельного коду. На вхід послідовного запису інформації DI поданий "0". Стрілками на діаграмі показано напрямок зчитування коду – від старшого розряду до молодшого. Як слідує з діаграм, після першого імпульсу на вході $C2$ в регістр був записаний код 0001, а після четвертого синхроімпульсу на вході $C1$ на виході $Q3$ (вихід послідовного коду) теж буде сформовано код 0001. Нескладно пересвідчитись, що після наступних синхроімпульсів на вході $C2$ процес перетворення паралельного коду в послідовний відбувається аналогічно.

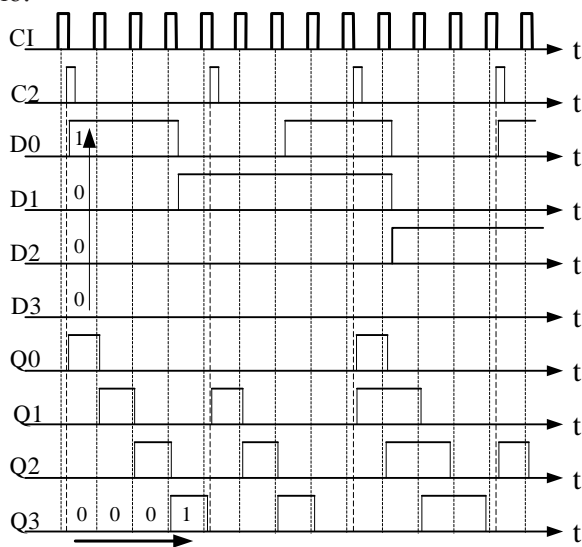


Рис. 10.16. Часові діаграми роботи перетворювача паралельного коду в послідовний

10.6.3. Перетворювач послідовного коду в паралельний

Для перетворення послідовного коду в паралельний можна використати два регістри (рис. 10.17). Перший регістр прий-

має послідовний код і після того, як він повністю буде введений, з виходів першого регістру він перезаписується у другий (паралельний) регістр, де і буде зберігатися до наступної зміни коду в першому регістрі.

Часові діаграми, що пояснюють роботу перетворювача послідовного коду в паралельний, наведені на рис. 10.18.

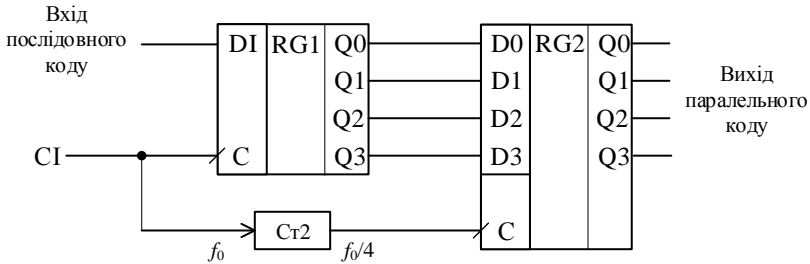


Рис. 10.17. Схема перетворювача послідовного коду в паралельний

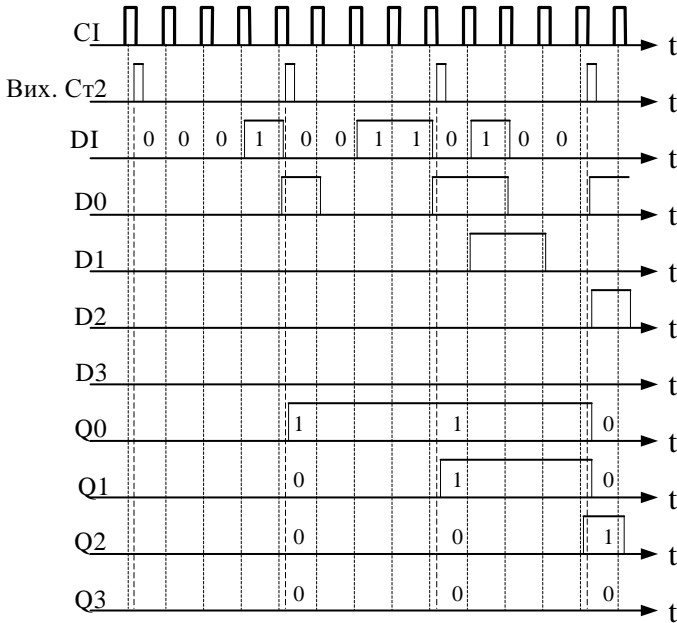


Рис. 10.18. Часові діаграми роботи перетворювача послідовного коду в паралельний

10.7. Проектування цифрового послідовнісного пристрою на регістрі зсуву

Спроекуємо генератор кодових послідовностей на регістрі зсуву для реалізації заданої кодової циклічної послідовності $N=(010111)$.

Спочатку необхідно визначити розрядність регістра:

010111010111

$n = 2$ $1 \rightarrow 2 \rightarrow 1 \rightarrow 3$

$n = 3$ $2 \rightarrow 5 \rightarrow 3 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 2$

$n = 4$ $5 \rightarrow 11 \rightarrow 7 \rightarrow 14 \rightarrow 13 \rightarrow 10 \rightarrow 5 \rightarrow 11$

При $n=4$ отримали замкнений цикл роботи пристрою, тобто основний перемикальний граф наведено на рис. 10.19.

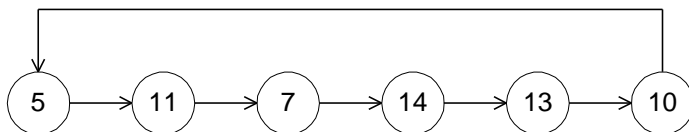


Рис. 10.19. Основний перемикальний граф роботи пристрою

В основному робочому циклі наявні 6 станів, тобто $M=6$. В цьому випадку кількість надлишкових (невикористовуваних) станів складає $M_H = 2^n - M = 2^4 - 6 = 16 - 6 = 10$.

Для перемикального графу (рис. 10.19) складемо діаграму термів, яку заповнюємо таким чином.

1. Якщо перехід з деякого стану відбувається до парного стану, то в клітинку на діаграмі термів, номер якої відповідає стану, з якого відбувається перехід, записують 0.

2. Якщо перехід з деякого стану відбувається до непарного стану, то в клітинку на діаграмі термів, номер якої відповідає стану, з якого відбувається перехід, записують 1.

3. До клітинок, номер яких відсутній на основному перемикальному графі, записують X (або 0, або 1).

Отримана для даного випадку діаграма термів наведена на рис. 10.20. На даній діаграмі необхідно об'єднати одиничні

клітинки з якомога більшою кількістю клітин, позначених символом X (при цьому клітини з номерами 0 та $2^n - 1$ (в нашому випадку 15) не можуть входити до об'єднань).

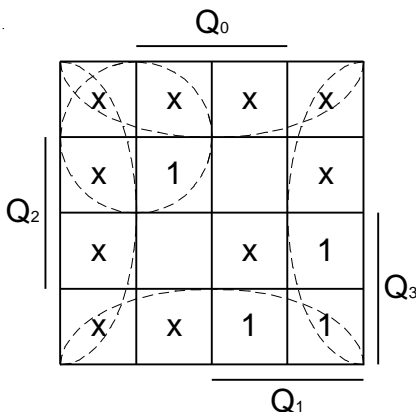


Рис. 10.20. Діаграма термів для генератора кодових послідовностей

Отримана при цьому функція зсуву буде мати такий вигляд:

$$D_0 = \overline{\overline{\overline{Q_2 + Q_0 + Q_1 \cdot Q_3}}} = \overline{\overline{Q_2 \cdot Q_0 \cdot Q_1 \cdot Q_3}}.$$

Далі проведемо аналіз функціонування пристрою, для чого спочатку перевіримо пристрій на самовідновність, визначивши наступні стани для кожного надлишкового стану за таким співвідношенням:

$$N^+ = \begin{cases} 2N + x_i, & N < 8 \\ 2(N - 2^{n-1}), & 8 \leq N < 16 \end{cases}$$

$$\begin{aligned} 0 &\rightarrow 1 \rightarrow 3 \rightarrow 7 \\ 2 &\rightarrow 5 \\ 4 &\rightarrow 9 \rightarrow 3 \\ 6 &\rightarrow 13 \\ 8 &\rightarrow 1 \\ 12 &\rightarrow 9 \\ 15 &\rightarrow 14 \end{aligned}$$

За результатами перевірки отримали, що з кожного надлишкового стану проєктований пристрій через одне або декілька перемикачів має повернутися до основного робочого циклу, тоб-

то самовідновність доведена. Тому можна побудувати повний перемикальний граф (рис. 10.21). Реалізована за функцією зсуву схема генератора кодових послідовностей наведена на рис. 10.22, а часові діаграми роботи пристрою – на рис. 10.23.

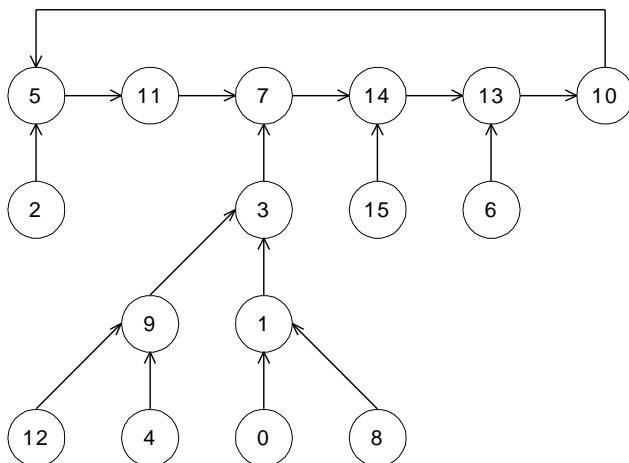


Рис. 10.21. Повний перемикальний граф

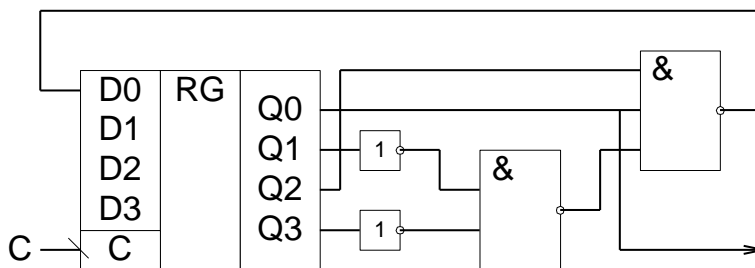


Рис. 10.22. Схема генератора кодових послідовностей

Для технічної реалізації схеми генератора виберемо дві мікросхеми серії КР1564ЛА1, що мають у своєму складі по 2 4-входових елементи І-НІ, та одну мікросхему серії SN74AS95N, що має у своєму складі 4-розрядний регістр зсуву, який перемикається з надходженням спаду синхроімпульсу.

Контрольні питання

1. таке регістр зсуву та які операції можна реалізувати за допомогою регістрів?

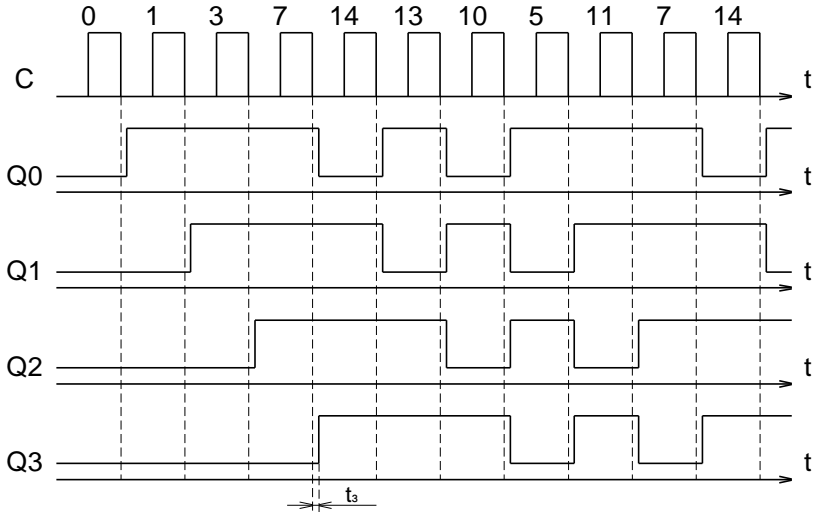


Рис. 10.23. Часові діаграми перемикання генератора кодових послідовностей

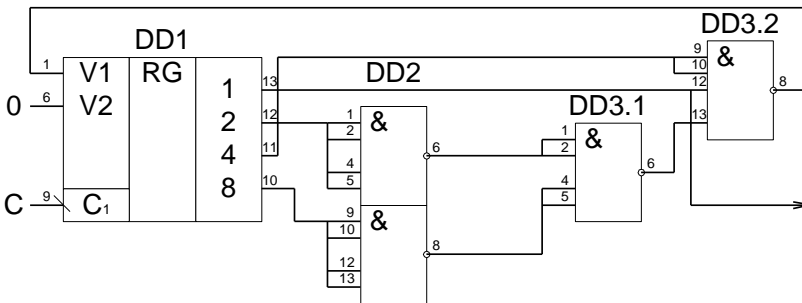


Рис. 10.24. Генератор кодових послідовностей на інтегральних мікросхемах

2. За якими ознаками можна провести класифікацію регістрів?

3. Які регістри потрібно використовувати для побудови перетворювачів послідовного коду в паралельний?

4. Які регістри потрібно використовувати для побудови перетворювачів паралельного коду в послідовний?

5. Які типи тригерів можна використовувати для побудови регістрів?

11. ЛІЧИЛЬНИКИ ІМПУЛЬСІВ

11.1 Призначення та класифікація лічильників імпульсів

Лічильники здійснюють рахунок і збереження коду числа підрахованих сигналів. Під сигналами маються на увазі імпульси або перепади напруги. Найпростішим лічильником є лічильний тригер, що здійснює рахунок і збереження не більше двох сигналів. З'єднуючи кілька тригерів певним чином, можна отримати лічильник з необхідним коефіцієнтом рахунку $K_{\text{рах}}$. Інша назва цього коефіцієнту – коефіцієнт ділення. Він показує, у скільки разів частота сигналу на виході лічильника буде нижчою за частоту сигналу на його вході, тобто $K_{\text{рах}} = f_{\text{вх}}/f_{\text{вих}}$.

Оскільки кожен тригер може знаходитися в одному з двох можливих станів, то лічильник, що складається з m тригерів, може мати $K_{\text{діл}} \leq 2^m$ станів. Перехід лічильника з одного стану до другого відбувається при подачі чергового вхідного сигналу.

Якщо перенумерувати послідовність вхідних сигналів від 0 до $K_{\text{діл}} - 1$, то кожному i -му номеру можна поставити у відповідність його двійковий еквівалент, виражений через стан лічильника, у який він перейде після приходу i -го вхідного сигналу. Визначаючи стан усіх тригерів лічильника за значеннями логічної змінної на їхніх прямих виходах, можна виразити кількість i підрахованих сигналів у вигляді m -розрядного двійкового коду.

Існують різні схеми лічильників, що відрізняються призначенням, типом використовуваних тригерів, організацією зв'язку між ними, порядком зміни станів, а також іншими особливостями.

Залежно від порядку зміни станів лічильники бувають з природним і довільним порядком рахунку. У перших значення коду кожного наступного стану відрізняється на 1 від попереднього, у других може відрізнятися більше, ніж на 1.

В свою чергу лічильники з природним порядком рахунку діляться на прості і реверсивні, а прості – на підсумовувальні та віднімальні. Реверсивні лічильники можуть працювати як в режимі додавання, так і в режимі віднімання.

За коефіцієнтом ділення лічильники діляться на двійкові ($K_{\text{діл}} = 2^m$) і недвійкові ($K_{\text{діл}} \neq 2^m$).

За способом перемикування тригерів під час роботи лічильники діляться на синхронні та асинхронні. В синхронних лічильниках всі тригери переключаються одночасно в момент надходження тактового імпульсу, а в асинхронних – після зміни стану на керуючому вході тригера.

Основними параметрами лічильників є їх ємність та швидкодія. Ємність лічильника характеризується $K_{\text{діл}}$, а швидкодія двома величинами:

- розділовою здатністю $t_p = 1 / f_{\text{вх}}$;

- часом встановлення $t_{\text{вст}}$ коду лічильника.

Розділова здатність визначається мінімально допустимим інтервалом часу між двома входними сигналами, при якому не відбувається втрата кількості підрахованих сигналів. Час встановлення коду $t_{\text{вст}}$ – інтервал часу між моментом часу надходження входного сигналу і моментом завершення переходу лічильника в новий стійкий стан.

Обов'язково потрібно пам'ятати, що код на виходах лічильника потрібно зчитувати тільки з прямих виходів тригерів. Це загальноприйнята умова, що дозволяє запобігати помилок при визначенні станів тригерів лічильника.

11.2. Двійкові лічильники

Основною ознакою двійкового лічильника є $K_{\text{діл}} = 2^m$. Для побудови таких лічильників можна використовувати різні види тригерів. Найбільше поширення отримали лічильники на основі універсальних JK-тригерів.

Характеристичне рівняння JK-тригера $Q^{t+1} = J' \bar{Q}' + \bar{K}' Q'$, де J' і K' – логічні функції входів J і K, що відповідають попередньому стану тригера.

Розглянемо підсумовувальний лічильник з $K_{\text{діл}} = 8$, при цьому кількість розрядів лічильника визначається за виразом $m = \log_2 K_{\text{діл}} = 3$.

Найпростішим за схемою є асинхронний підсумовувальний лічильник на Т-тригерах. Оскільки окремо Т-тригери не випускаються, використаємо JK-тригери, що працюють в режимі Т-тригера. Схема лічильника з $K_{\text{діл}} = 8$ наведена на рис. 11.1, а часові діаграми сигналів, що пояснюють його роботу – на рис. 11.2.

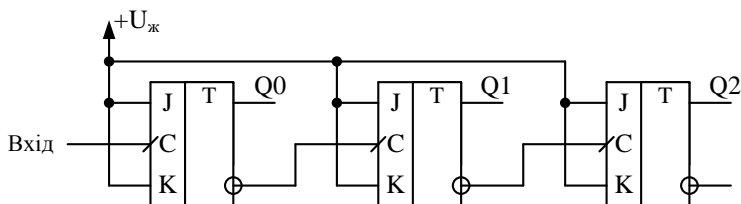


Рис. 11.1. Схема трирозрядного асинхронного двійкового підсумовувального лічильника

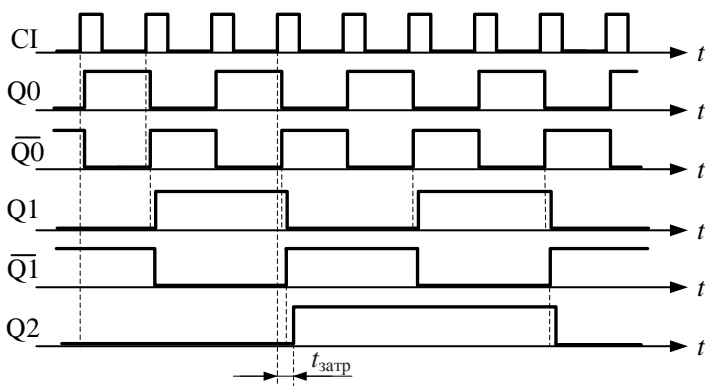


Рис. 11.2. Часові діаграми роботи трирозрядного асинхронного двійкового підсумовувального лічильника

Оскільки JK-тригер працює в режимі лічильного тригера за умови, що на його входах J і K встановлена "1", то ці входи тригерів (рис. 11.1) під'єднані до джерела живлення. Аналізуючи діаграми роботи такого лічильника, неважко скласти таблицю станів тригерів лічильника (табл. 11.1). З таблиці і часових діаг-

рам виходить, що значення коду на прямих виходах під час приходу кожного наступного синхроімпульсу збільшується, а повний цикл повторення складається з восьми імпульсів. З цього можна зробити висновок, що коефіцієнт ділення дорівнює 8, а лічильник є підсумовувальним.

Таблиця 11.1. Зміна станів на виходах підсумовувального лічильника

n	Q0	Q1	Q2
1	0	0	0
2	1	0	0
3	0	1	0
4	1	1	0
5	0	0	1
6	1	0	1
7	0	1	1
8	1	1	1
9	0	0	0

Час затримки спрацювання останнього тригера лічильника можна розрахувати за простою формулою $t_{затр} = t_{затр1} \cdot n$, де $t_{затр1}$ – затримка спрацювання одного тригера по відношенню до моменту приходу синхроімпульсу на його вхід С (рис. 11.2).

Якщо під'єднати до входів С кожного наступного тригера не інверсні, а прямі виходи попередніх тригерів (рис. 11.3), то отримаємо віднімальний асинхронний лічильник, часові діаграми роботи якого наведені на рис. 11.4.

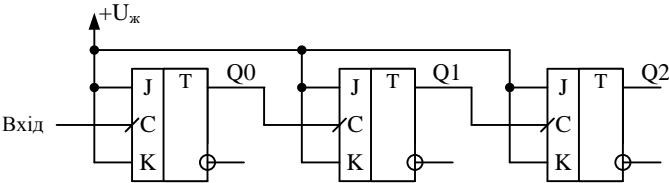


Рис. 11.3. Схема трирозрядного асинхронного двійкового віднімального лічильника

Як слідує з часових діаграм, кожний вхідний імпульс ви-

кликає зменшення вихідного коду на виходах тригерів на одиницю, тобто такий лічильник є віднімальним.

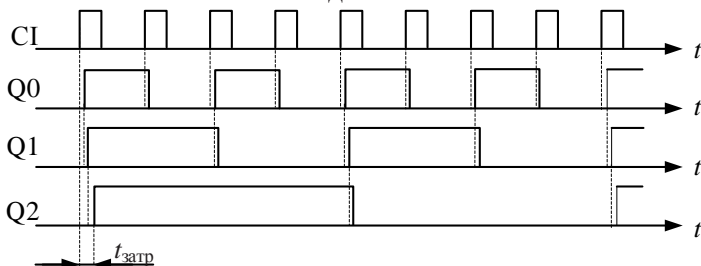


Рис. 11.4. Часові діаграми роботи трирозрядного асинхронного двійкового віднімального лічильника

11.3. Реверсивні лічильники

Реверсивними називають лічильники, що можуть працювати або як підсумовуючий, або як віднімаючий, в залежності від значення сигналу управління.

Розглянемо принцип побудови такого лічильника на базі трьохрозрядного двійкового лічильника. Функціональна схема такого лічильника наведена на рис. 11.5.

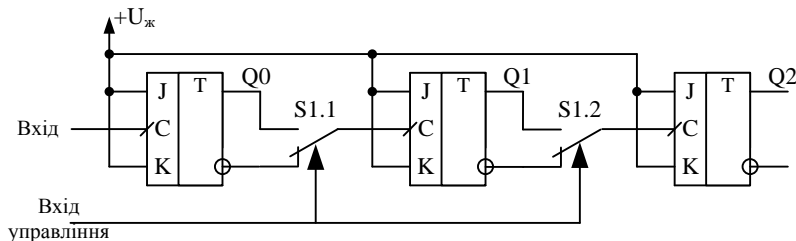


Рис. 11.5. Функціональна схема трирозрядного асинхронного реверсивного лічильника

З аналізу роботи схем, які наведені на рис. 11.3 та 11.5, можна зробити висновок, що якщо ключі S1 встановлені у нижнє положення (рис. 11.5), то лічильник буде працювати як підсумовувальний, а якщо у верхнє – як віднімальний.

Замінивши перемикачі електронними ключами, отримаємо трирозрядний реверсивний лічильник, схема якого наведена

на рис. 11.6.

При подачі на вхід управління нульового рівня $V="0"$ лічильник буде працювати як підсумовувальний, оскільки сигнали з інверсних виходів тригерів будуть передаватись на входи синхронізації наступних тригерів, а при $V="1"$ – як віднімальний.

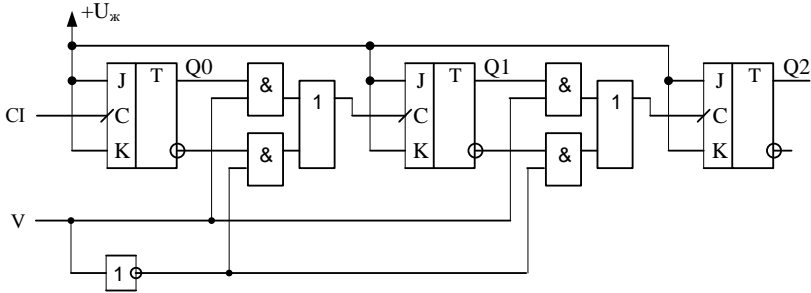


Рис. 11.6. Схема триризрядного асинхронного реверсивного лічильника

11.4. Синхронні двійкові лічильники

Для зменшення часу перемикавання лічильника використовуються синхронні лічильники, у яких всі тригери перемикаються одночасно. Найбільш зручною виявляється реалізація синхронних лічильників на JK-тригерах.

Характеристичне рівняння синхронного лічильника на JK-тригерах має вигляд

$$Q^{t+1} = (J_1 J_2 J_3 J_4)^t \bar{Q}^t + (\bar{K}_1 \bar{K}_2 \bar{K}_3 \bar{K}_4)^t Q^t = J^t \bar{Q}^t + \bar{K}^t Q^t.$$

На рис. 11.7 наведена схема чотириризрядного синхронного двійкового лічильника, побудованого на універсальних RSJK-тригерах з інверсними входами R та S. Оскільки входи R та S під'єднані до джерела живлення, то на роботу тригерів вони не впливають.

Логічні елементи І потрібні для того, щоб перемикавання кожного наступного тригера відбувалось тільки після перемикавання всіх попередніх тригерів. При побудові часових діаграм роботи лічильника треба пам'ятати, що перемикавання JK-тригерів відбувається тільки якщо під час надходження синхроімпульсу на входах J і K встановлений рівень лог. 1.

Часові діаграми сигналів у різних точках лічильника наведені на рис. 11.8. Як слід з часових діаграм, сигнал на виході першого логічного елементу І (точка А на схемі) формується тільки після того, як переключилися і перший, і другий тригери. Аналогічно формується сигнал у точці В з урахуванням спрацьовування перших трьох тригерів лічильника.

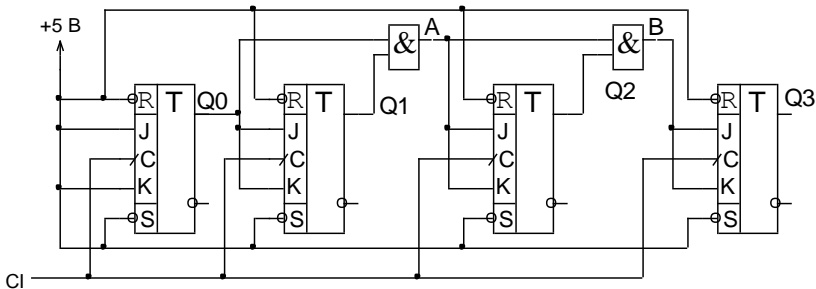


Рис. 11.7. Схема чотирирозрядного синхронного лічильника

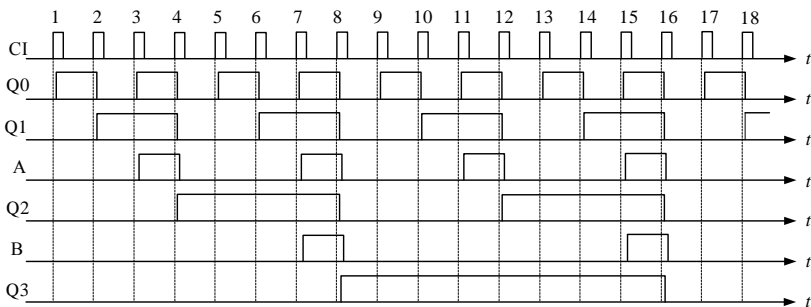


Рис. 11.8. Часові діаграми роботи чотирирозрядного синхронного лічильника

11.5. Недвійкові лічильники

11.5.1. Загальні зауваження

Основною ознакою, що лічильник недвійковий, є виконання умови $K_{\text{діл}} \neq 2^n$. Принцип побудови недвійкових лічильників полягає у виключенні деяких стійких станів звичайного двійкового лічильника, що є надлишковими для недвійкового лі-

чильника. Надлишкові стани виключаються за допомогою зворотних зв'язків у лічильнику. Зворотні зв'язки утворюють введенням додаткових логічних елементів, що з'єднують входи і виходи відповідних тригерів.

Основні способи побудови недвійкових лічильників (або лічильників з довільним коефіцієнтом ділення):

- синтез лічильника на рівні логічних рівнянь з мінімізацією кількості кіл зворотного зв'язку;
- використання готового лічильника і введення до його схеми зворотних зв'язків для виключення надлишкових станів;
- використання програмованих лічильників для формування потрібного коефіцієнта ділення;
- використання готового лічильника і введення до його схеми кола встановлення до початкового стану, що складається з дешифратора та мультиплексора.

11.5.2. Синтез лічильника із заданим коефіцієнтом ділення

Задача синтезу недвійкового лічильника зводиться до визначення необхідних зворотних зв'язків та мінімізації їх кількості.

Кількість тригерів у недвійковому лічильнику визначається з виразу

$$n = \lceil \log_2 K_{\text{діл}} \rceil,$$

де $\lceil \rceil$ – знак округлення до найближчого цілого числа.

Кількість надлишкових станів, що виключаються:

$$N = 2^n - K_{\text{діл}}.$$

Оскільки можна виключити будь-які стани в будь-яких комбінаціях, то загальна кількість схем недвійкового лічильника з тим самим $K_{\text{діл}}$ та усіма варіантами зміни порядку рахунку визначається величиною

$$m = \frac{(K_{\text{діл}} - 1)! K_{\text{діл}}}{N! (K_{\text{діл}} - N)}.$$

Розглянемо процес синтезу лічильника на прикладі побудови синхронного лічильника з коефіцієнтом ділення $K_{\text{діл}} = 3$.

Для цього потрібно виконати такі кроки.

Крок №1. Знаходимо необхідну кількість тригерів для побудови лічильника:

$$n = \lceil \log_2 K_{\text{дл}} \rceil = \lceil \log_2 3 \rceil = 1.58 \approx 2.$$

Отже, лічильник потрібно будувати на основі двійкового синхронного лічильника з $K_{\text{дл}} = 4$, що складається з двох тригерів.

Крок №2. Знаходимо кількість надлишкових станів:

$$N = 2^n - K_{\text{дл}} = 2^2 - 3 = 1.$$

Лічильник на двох тригерах може мати $2^2 = 4$ стани:

$$Q_1 Q_2; \overline{Q_1} Q_2; Q_1 \overline{Q_2}; \overline{Q_1} \overline{Q_2}.$$

Виключимо, наприклад, стан $Q_1 Q_2$, а порядок зміни станів приймемо такий:

$$Q_1 Q_2; Q_1 \overline{Q_2}; \overline{Q_1} Q_2; \overline{Q_1} \overline{Q_2}.$$

Крок №3. Переходимо до синтезу лічильника. Побудуємо лічильник на JK-тригерах. Характеристичне рівняння роботи JK-тригера:

$$Q^{t+1} = J^t \overline{Q}^t + \overline{K}^t Q^t.$$

Складемо таблицю переходів JK-тригера (табл. 11.2).

Таблиця 11.2. Таблиця переходів JK-тригера

J^t	K^t	Q^t	Q^{t+1}
0	0	0	0
0	1	0	0
1	0	0	1
1	1	0	1
0	0	1	1
1	0	1	1
0	1	1	0
1	1	1	0

Крок №4. Складаємо таблицю функціонування лічильника (таблиця 11.3) та сформуємо прикладні таблиці для кожного з тригерів (рис.11.9).

У клітинках, де стоїть знак "-" або "*", функція є неви-

значеною.

Таблиця 11.3. Порядок зміну станів лічильника

Q	Номер стану		
	1 $\overline{Q_1}\overline{Q_2}$	2 $Q_1\overline{Q_2}$	3 $\overline{Q_1}Q_2$
Q_1^t	0	1	0
Q_2^t	0	0	1
Q_1^{t+1}	1	0	0
Q_2^{t+1}	0	1	0

$Q_1^t \longrightarrow$	$\overline{Q_2^t}$	Q_2^t
	$\overline{Q_1^t}$	Q_1^t
	01	00
	10	--
	$\overline{Q_2^t}$	Q_2^t
	$\overline{Q_1^t}$	Q_1^t
	00	10
	01	--

Рис. 11.9. Прикладні таблиці тригерів лічильника

Крок №5. На основі таблиці переходів JK-тригера складемо характеристичну таблицю даного тригера.

Таблиця 11.4. Характеристична таблиця JK-тригера

$Q^t \rightarrow Q^{t+1}$	J^t	K^t
00	0	*
01	1	*
10	*	1
11	*	0

Крок №6. На основі прикладних та характеристичної та-

блиць складемо карти Карно для входів J і K тригерів лічильника.

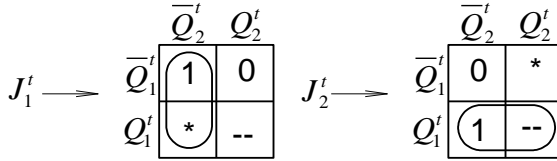


Рис. 11.10. Карти Карно для J-входів тригерів

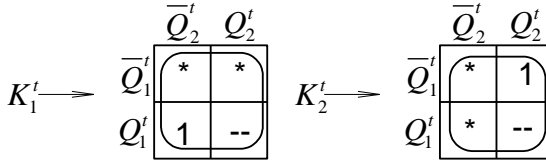


Рис. 11.11. Карти Карно для K-входів тригерів

Використовуючи принцип мінімізації і карти Карно, отримуємо такий результат:

$$\begin{aligned} J_1^t &= \overline{Q}_2^t \overline{Q}_1^t + \overline{Q}_2^t Q_1^t = \overline{Q}_2^t (\overline{Q}_1^t + Q_1^t) = \overline{Q}_2^t, \\ J_2^t &= \overline{Q}_2^t Q_1^t + Q_2^t Q_1^t = Q_1^t (\overline{Q}_2^t + Q_2^t) = Q_1^t, \\ K_1^t &= K_2^t = Q_1^t Q_2^t + \overline{Q}_1^t Q_2^t + Q_1^t \overline{Q}_2^t + \overline{Q}_1^t \overline{Q}_2^t = \\ &= Q_1^t (Q_2^t + \overline{Q}_2^t) + \overline{Q}_1^t (\overline{Q}_2^t + Q_2^t) = Q_1^t + \overline{Q}_1^t = 1. \end{aligned}$$

Таким чином, для побудови лічильника з коефіцієнтом ділення $K_{\text{діл}}=3$ необхідно вхід J_1 1-го тригера з'єднати з виходом \overline{Q}_2 , J_2 – з виходом Q_1 ; а на входи K_1 і K_2 подати сигнал лог. 1.

На основі попереднього аналізу отримуємо синтезовану схему (рис. 11.12).

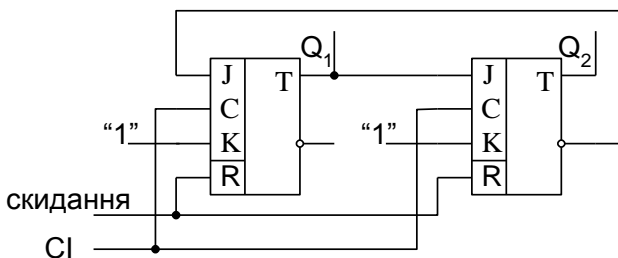


Рис. 11.12. Схема синтезованого лічильника з коефіцієнтом ділення 3

11.5.3. Побудова недвійкового лічильника з використанням готового лічильника

Приймемо за основу схему синхронного лічильника, що наведена на рис. 11.7. Для формування потрібного коефіцієнта ділення використаємо входи встановлення до "0" всіх тригерів, тобто введемо зворотний зв'язок з виходу додаткового логічного елемента (рис. 11.13) на входи R.

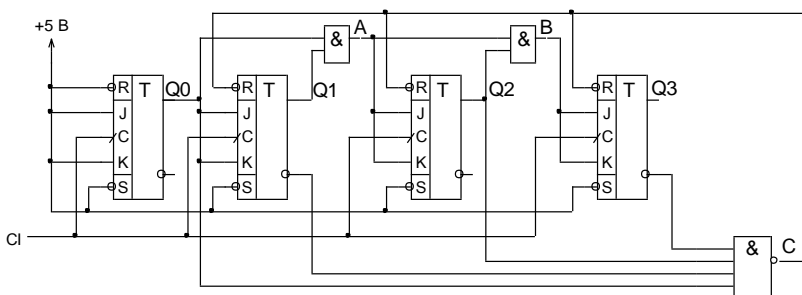


Рис. 11.13. Схема лічильника з коефіцієнтом ділення 5

Коефіцієнт ділення такого лічильника чисельно буде дорівнювати коду на виходах тригерів, при якому на виході третього логічного елемента (вихід C) буде сформовано імпульс.

Для того, щоб у точці C було сформовано імпульс, необхідно, щоб на входах елемента I були встановлені рівні лог. 1, а ця умова буде виконуватись при $Q_3 = 0$, $Q_2 = 1$, $Q_1 = 0$, $Q_0 = 1$, тобто на виходах лічильника встановлюється код 1010, що відповідає десятковому значенню 5.

Часові діаграми, що пояснюють роботу такого лічильни-

ка, наведені на рис. 11.14.

В момент часу, коли на виходах лічильника встановлюється код 0101, в точці С формується лог. 0 і перший та третій тригер встановлюються до стану лог. 0. Одразу після цього умова існування сигналу в точці С порушується і на виході логічного елемента встановлюється знову рівень лог. 1. Тривалість імпульсу у точці С буде дорівнювати сумі затримок, що вносять логічний елемент І та тригери при перемиканні до стану лог. 0.

При підрахунку коефіцієнта ділення враховуються тільки ті періоди, під час яких стан на виходах тригерів (після перемикання під дією синхроімпульсу) не змінюється.

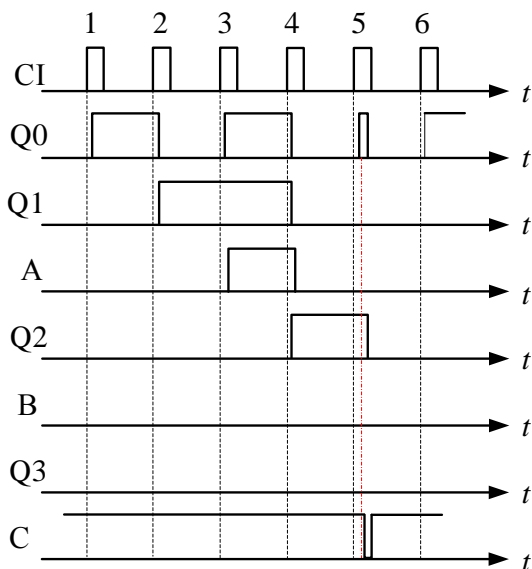


Рис. 11.14. Часові діаграми роботи лічильника з коефіцієнтом ділення 5

Якщо для формування потрібного коефіцієнта ділення використати входи S тригерів, то коефіцієнт ділення буде дорівнювати $N+1$, де N – значення коду, при якому на виході С формується сигнал.

Звісно, що для побудови лічильників таким способом можна використовувати й інші логічні елементи та тригери з прямими входами R та S.

11.5.4. Використання програмованих лічильників для побудови лічильників з довільним коефіцієнтом ділення

Програмованим називають лічильник, в якому передбачені додаткові входи для запису початкового коду в тригери лічильника. Потрібний коефіцієнт ділення формується дорахуванням від встановленого до кінцевого значення коду лічильника. Під кінцевим значенням коду найчастіше розуміють встановлення на виході лічильника нульового значення коду.

Тоді коефіцієнт ділення для підсумовувального лічильника буде дорівнювати

$$K_{\text{діл}} = K_{\text{макс}} - N, \quad (11.1)$$

де $K_{\text{макс}} = 2^n$, n – кількість тригерів лічильника, N – значення початкового коду, записаного до лічильника.

Для віднімального лічильника коефіцієнт ділення чисельно дорівнює значенню коду, записаного у лічильник: $K_{\text{діл}} = N$.

Принцип побудови програмованих лічильників досить простий. Якщо використовувати універсальні RSJK-тригери, то входи R та S можна використати для програмування, як показано на рис. 11.15, б.

Для програмування необхідно подати на вхід даних (D0 на рис. 11.15, б) потрібне значення коду цього розряду і подати на вхід L (Load – завантаження) сигнал лог. 1 (короткий імпульс). Якщо значення D0=1, то на вході S тригера встановиться лог. 1, а на вході R – лог. 0, оскільки на верхній вхід логічного елементу I2 сигнал з входу D0 подається через інвертор. Тому тригер встановиться до стану лог. 1.

Якщо на вхід D0 подати лог. 0, то рівень лог. 1 встановиться на вході R, а тригер встановиться до стану лог. 0.

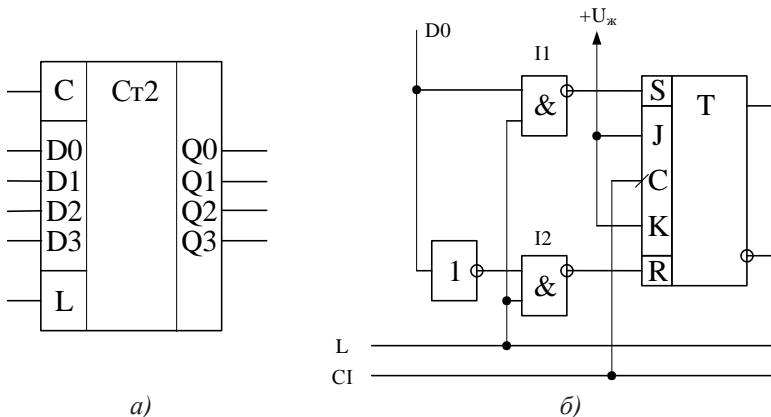


Рис. 11.15. Умовне графічне позначення програмованого лічильника (а) та схема одного з його розрядів (б)

Більшість лічильників, що випускаються промисловістю, забезпечуються додатковими входами для попереднього запису паралельного коду. Розглянемо найбільш поширений реверсивний програмований двійковий лічильник КР1533ІЕ7 (аналог мікросхеми 74LS193), що випускається у складі багатьох серій ІМС ТТЛ та КМОН. Аналогічний за функціями двійково-десятковий лічильник КР1533ІЕ6 (74LS192).

Умовне графічне позначення мікросхеми КР1533ІЕ7 наведено на рис. 11.16, а, а умовні графічні позначення 74LS193 наведено на рис. 11.16, б та в. У довідниках можна знайти ще декілька різних позначень зазначених мікросхем.

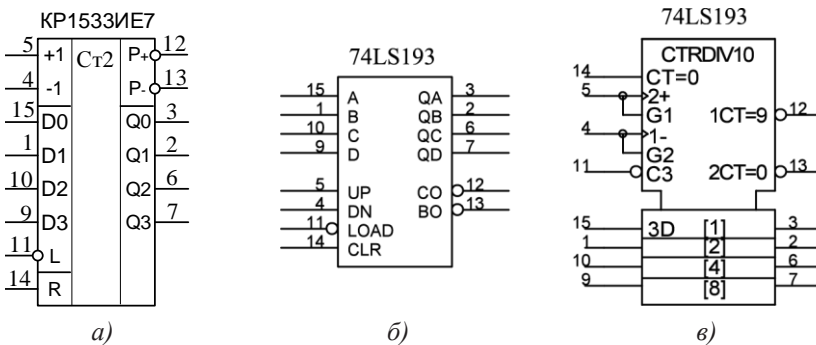


Рис. 11.16. Умовне графічне позначення програмованого лічильника

KP1533IE7 (a) та його аналогу 74LS193 (б, в)

Запис коду початкової установки здійснюється при подачі на вхід дозволу запису L (*Load* – завантаження) активного рівня та занесенні до кожного розряду лічильника інформації, що подається на входи програмування D0...D3.

Оскільки цей лічильник реверсивний, то в ньому передбачено два входи для зміни напрямку роботи: +1 та -1. Для того, щоб лічильник працював у режимі додавання, необхідно на вхід +1 подати імпульси, а на вхід -1 – рівень лог. 1. Для зміни напрямку лічби слід зробити навпаки. На рис. 11.17 наведені часові діаграми сигналів, що ілюструють роботу лічильника в різних режимах, а у табл. 11.5 – значення сигналів на входах мікросхеми в різних режимах роботи.

Імпульси перенесення лічильника, що формуються на виходах P+ в режимі додавання та P- в режимі віднімання, є затриманим вхідним імпульсом.

Таблиця 11.5. Таблиця функціонування лічильника KP1533IE7

Режим роботи	Входи			
	R	L	+1	-1
Скидання до лог. 0	1	X	X	X
Завантаження коду	0	0	X	X
Додавання	0	1	0→1	1
Віднімання	0	1	1	0→1

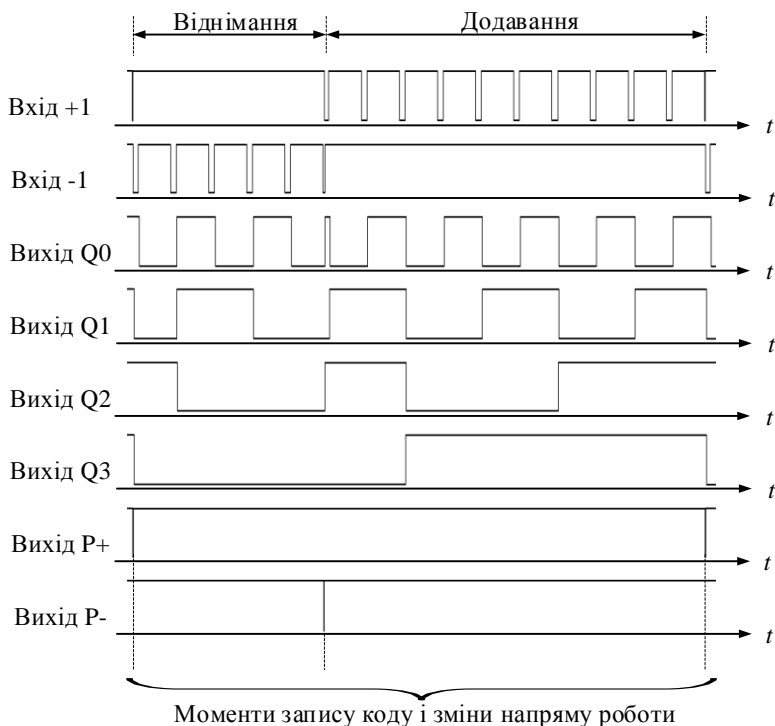


Рис. 11.17. Часові діаграми роботи лічильника KP1533IE7

Імпульси перенесення можна використовувати для з'єднання декількох лічильників між собою для збільшення розрядності (рис. 11.18), а також для програмування лічильника.

Для програмування лічильника на його вхід завантаження (L) потрібно подати імпульс з виходу перенесення P+ (якщо лічильник працює в режимі додавання), або з виходу P- (якщо лічильник працює в режимі віднімання).

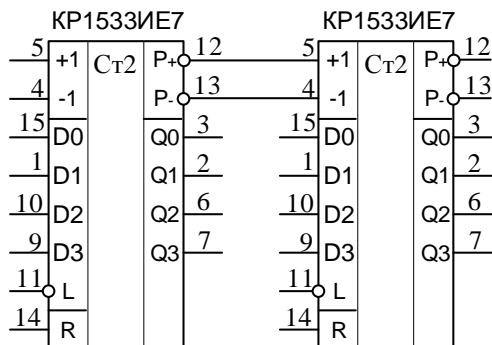


Рис. 11.18. З'єднання лічильників KP1533IE7 для збільшення розрядності

Проілюструвати використання програмованих лічильників можна на прикладі побудови лічильника з потрібним коефіцієнтом лічби, що не дорівнює 2^n та 10^n .

Розглянемо приклад. Нехай коефіцієнт ділення лічильника, що працює в режимі додавання коду, потрібно реалізувати рівним 237. Для побудови такого лічильника потрібно використати дві мікросхеми KP1533IE7. Тоді максимально можливий коефіцієнт ділення буде дорівнювати 256.

Знаходимо код програмування з співвідношення (11.1):

$$N = K_{\text{макс}} - K_{\text{діл}} = 256 - 237 = 19.$$

Переводимо отримане значення N з десятикової системи числення у двійковий 8-розрядний код, при цьому отримаємо $N=00010011_2$. Відповідна схема лічильника з $K_{\text{діл}} = 237$ наведена на рис. 11.19.

Код на входах програмування задається підключенням до спільного проводу та до резистора R1. В першому лічильнику потрібно встановити лог. 1 на входах молодших розрядів D0 та D1, а в другому лічильнику – на вході D0. Як тільки на виході перенесення P+ буде сформовано імпульс, у тригери лічильника буде записаний код 00010011 і почнеться відлік коду від цього значення. Після завершення циклу роботи (при переході тригерів лічильника з стану 11111111 до стану 00000000) на виході P+ знову буде сформовано імпульс перенесення і весь процес буде повторюватись.

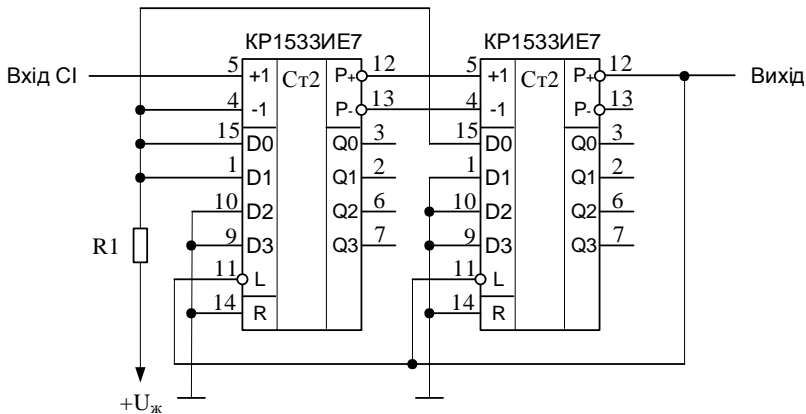


Рис. 11.19. Схема підсумовувального лічильника з $K_{\text{діл}} = 237$

11.5.5. Побудова недвійкових лічильників за допомогою дешифраторів

Недвійкові лічильники можна побудувати, використавши для встановлення коефіцієнта ділення дешифратори двійкового коду. Сутність способу побудови таких лічильників полягає у тому, що вхід встановлення лічильника до стану лог. 0 під'єднується до одного з виходів дешифратора, входи якого підключені до виходів лічильника. Як тільки на виході лічильника буде встановлений код, що відповідає сигналу на використаному виході дешифратора, лічильник переходить до початкового стану, тобто на всіх його виходах встановлюється значення лог. 0 і весь процес повторюється знову.

На рис. 11.20 наведена схема лічильника, коефіцієнт ділення якого може встановлюватися в межах від 1 до 16.

Розглянемо роботу схеми при встановленні перемикача S1 у перше положення, як показано на рис. 11.20. Після встановлення лічильника до стану лог. 0 код на його виході починає змінюватись при надходженні на вхід чергового синхроімпульсу. Стан лічильника буде змінюватись таким чином: 0000, 0001. Як тільки на виході лічильника встановиться код 0001, на виході Q1 дешифратора буде сформовано сигнал рівня лог. 1, який через перемикач S1 подається на вхід R лічильника і встановлює

його знову у початковий стан. Таким чином, після першого синхроїмпульсу весь процес повторюється. Відповідно коефіцієнт ділення лічильника буде дорівнювати 1.

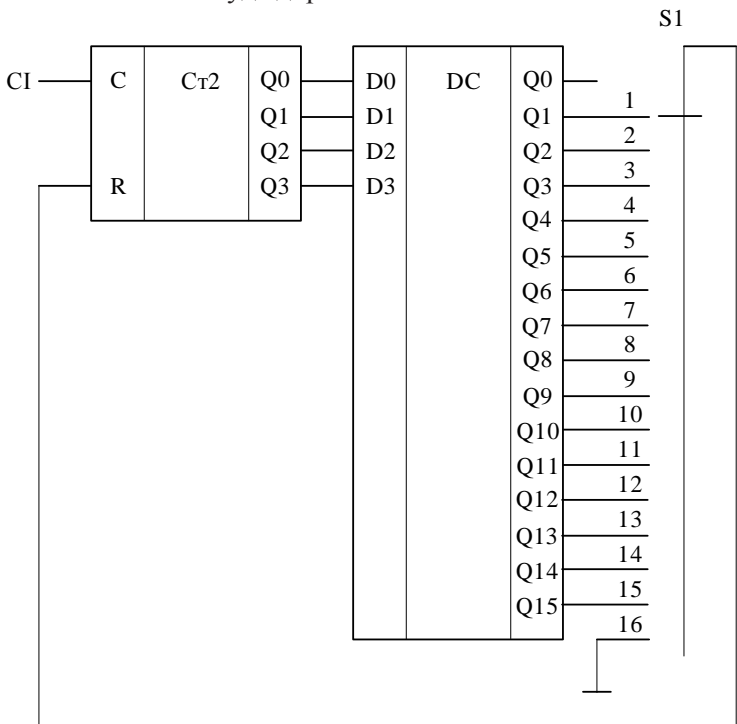


Рис. 11.20. Схема лічильника зі змінним коефіцієнтом ділення

Якщо перевести перемикач до положення 16, то на вхід R лічильника сигнали встановлення до стану лог. 0 надходити не будуть і коефіцієнт ділення буде складати 16. Незавжди побажити, що нумерація виводів перемикача та виходів дешифратора відповідають значенню коефіцієнта ділення, що буде встановлений перемикачем у кожному з цих положень.

11.6. Лічильники на регістрах зсуву

При побудові таких лічильників необхідно на початку роботи здійснювати спеціальне кодування їхніх внутрішніх ста-

нів. Для цього потрібно записати до такого лічильника будь-яке значення коду (окрім всіх нулів або одиниць).

Такі лічильники отримали назву кільцевих лічильників або лічильників Джонсона. Якщо на вхід СІ (рис. 11.21) подати періодичну послідовність імпульсів, то внутрішні стани регістра зсуву будуть також періодично повторюватися, тобто регістр зсуву буде являти собою лічильник за деяким $\text{mod } M$, якщо зазначена періодична послідовність сформована самим регістром зсуву. З цього випливає, що кодування внутрішніх станів, наприклад, лічильника за $\text{mod } 5$, може бути задано схемою 00011. Для побудови схеми будемо використовувати функцію збудження $\overline{D0} = \overline{Q_2} \overline{Q_3}$ і отримаємо схему, наведену на рис. 11.21.

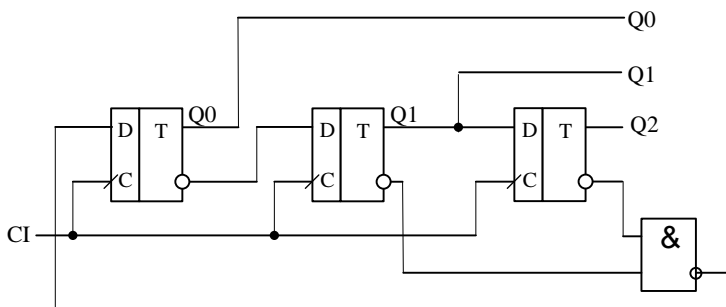


Рис. 11.21. Схема кільцевого лічильника з $K_{dil} = 5$

При проходженні через регістр, що складається з трьох тригерів, такої періодичної послідовності символів 0 і 1, отримаємо п'ять різних кодових комбінацій (внутрішніх станів): 000, 010, 111, 101, 001 (рис. 11.22). При подальшому зсуві формуються такі самі кодові комбінації, тому що послідовність символів 0 і 1 періодична.

При кодуванні внутрішніх станів лічильників за допомогою періодичної послідовності символів 0 і 1 варто керуватися такими правилами:

- кількість символів M в одному періоді послідовності визначає коефіцієнт ділення лічильника;
- мінімальну кількість тригерів m знаходять за умови отримання при зсуві періодичної послідовності символів 0 та 1 M різних кодових комбінацій.

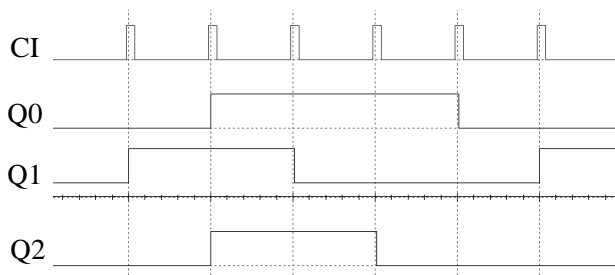


Рис. 11.22. Часові діаграми кільцевого лічильника з $K_{011} = 5$

11.7. Застосування регістрів та лічильників для побудови схем затримки імпульсних сигналів

Використовуючи регістри зсуву, можна побудувати схему дискретної затримки імпульсних сигналів. Такі задачі виникають при формуванні сигналів керування різними пристроями.

На рис. 11.23 наведена функціональна схема такого пристрою.

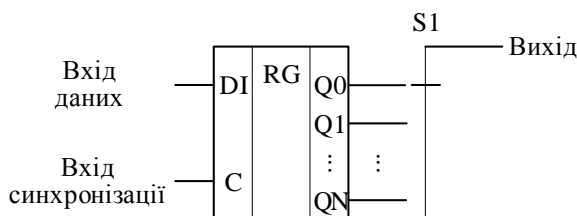


Рис. 11.23. Функціональна схема пристрою дискретної затримки імпульсних сигналів

Вхідні дані записуються до першого тригера регістра зсуву через вхід DI (*Data Input*) і з кожним тактовим імпульсом, що надходить на вхід синхронізації, просуваються по регістру на один розряд.

Якщо період слідування синхроімпульсів позначити як T_{Cl} , а вихідний сигнал знімати з n -го виходу регістра за допомогою перемикача S1, то час затримки буде дорівнювати nT_{Cl} .

Для того, щоб побудувати пристрій дискретної затримки,

необхідно задати максимальний час t_{\max} і крок зміни затримки Δt_3 . Тоді кількість розрядів регістра можна знайти за формулою $N = t_{\max}/\Delta t_3$.

Якщо необхідна кількість розрядів регістра для організації необхідної затримки є великою, то можна побудувати схему затримки з використанням декількох регістрів, кожний з яких дозволяє змінювати затримку на різний час. Тоді функціональна схема такого пристрою буде мати такий вигляд (рис. 11.24).

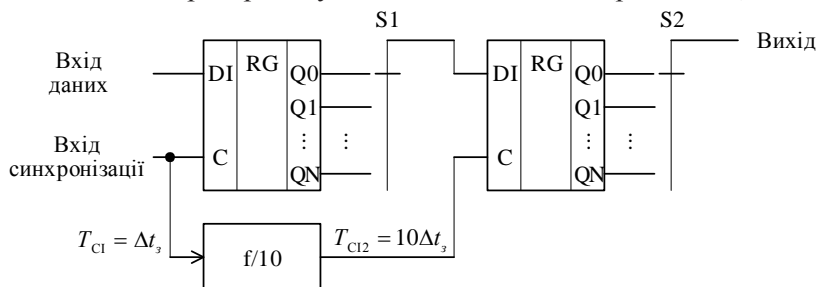


Рис. 11.24. Функціональна схема пристрою двоступеневої дискретної затримки імпульсних сигналів

Схема складається з двох регістрів та подільника частоти з коефіцієнтом ділення 10. У такій схемі за допомогою перемикача S1 можна дискретно змінювати затримку з кроком Δt_3 , а за допомогою S2 – з кроком $10\Delta t_3$. Якщо змінити коефіцієнт ділення подільника частоти, то дискретність перебудови часу затримки другого ступеня можна зменшити. Кількість таких ступенів може бути довільною.

Використання багатоступінчатих схем затримки дозволяє значно скоротити необхідну кількість розрядів регістра.

11.8. Проектування недвійкових лічильників на тригерах та мікросхемах

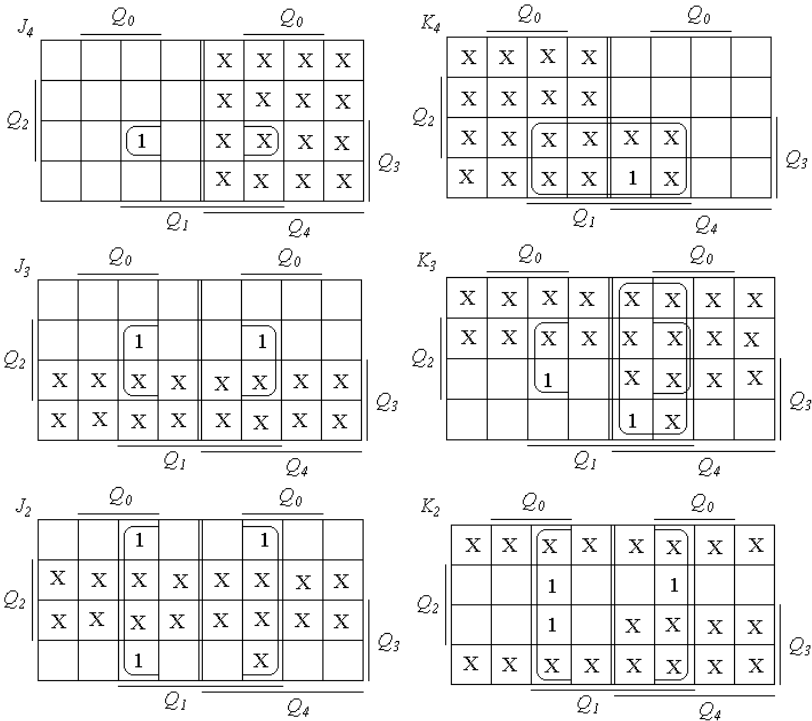
11.8.1. Проектування паралельного реверсивного лічильника на JK-тригерах

Спроектуємо лічильник паралельний реверсивний на JK-тригерах з коефіцієнтом перерахунку (модулем лічби) $M=27$.

Визначаємо розрядність пристрою, виходячи з нерівності $2^{n-1} < M \leq 2^n$, звідки $n=5$.

Кількість надлишкових станів: $M_H = 2^n - M = 5$.

Складаємо перемикальну таблицю лічильника (табл. 11.6), за якою заповнюємо діаграми термів для кожного входу керування тригерів і визначаємо функції збудження тригерів.



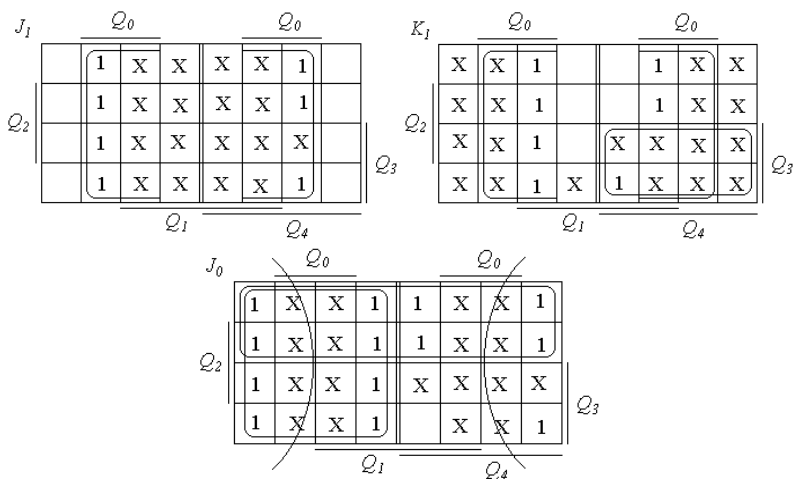
$$J_4 = Q_3 Q_2 Q_1 Q_0; \quad K_4 = Q_3 Q_1;$$

$$J_3 = Q_2 Q_1 Q_0; \quad K_3 = Q_2 Q_1 Q_0 + Q_4 Q_1;$$

$$J_2 = Q_1 Q_0; \quad K_2 = Q_1 Q_0.$$

Таблиця 11.6. Перемикальна таблиця лічильника (М=27)

N	$Q_4 Q_3 Q_2 Q_1 Q_0$	$Q_4^+ Q_3^+ Q_2^+ Q_1^+ Q_0^+$	$J_4 K_4$	$J_3 K_3$	$J_2 K_2$	$J_1 K_1$	$J_0 K_0$
0	00000	00001	0X	0X	0X	0X	1X
1	00001	00010	0X	0X	0X	1X	X1
2	00010	00011	0X	0X	0X	X0	1X
3	00011	00100	0X	0X	1X	X1	X1
4	00100	00101	0X	0X	X0	0X	1X
5	00101	00110	0X	0X	X0	1X	X1
6	00110	00111	0X	0X	X0	X0	1X
7	00111	01000	0X	1X	X1	X1	X1
8	01000	01001	0X	X0	0X	0X	1X
9	01001	01010	0X	X0	0X	1X	X1
10	01010	01011	0X	X0	0X	X0	1X
11	01011	01100	0X	X0	1X	X1	X1
12	01100	01101	0X	X0	X0	0X	1X
13	01101	01110	0X	X0	X0	1X	X1
14	01110	01111	0X	X0	X0	X0	1X
15	01111	10000	1X	X1	X1	X1	X1
16	10000	10001	X0	0X	0X	0X	1X
17	10001	10010	X0	0X	0X	1X	X1
18	10010	10011	X0	0X	0X	X0	1X
19	10011	10100	X0	0X	1X	X1	X1
20	10100	10101	X0	0X	X0	0X	1X
21	10101	10110	X0	0X	X0	1X	X1
22	10110	10111	X0	0X	X0	X0	1X
23	10111	11000	X0	1X	X1	X1	X1
24	11000	11001	X0	X0	0X	0X	1X
25	11001	11010	X0	X0	0X	1X	X1
26	11010	00000	X1	X1	0X	X1	0X
27-31	11XXX	XXXXX	XX	XX	XX	XX	XX



$$J_1 = Q_0; \quad K_1 = Q_0 + Q_3 Q_4; \quad J_0 = \overline{Q_3 Q_4 Q_1}; \quad K_0 = 1.$$

Далі проведемо аналіз функціонування пристрою та доведемо його самовідновність.

Доведення самовідновності:

$$N = 11011_2 = 27_{10}$$

$J_0 = 0$	$K_0 = 1$	$Q_0^+ = 0$
$J_1 = 1$	$K_1 = 1$	$Q_1^+ = 0$
$J_2 = 1$	$K_2 = 1$	$Q_2^+ = 1$
$J_3 = 0$	$K_3 = 1$	$Q_3^+ = 0$
$J_4 = 0$	$K_4 = 1$	$Q_4^+ = 0$

$$N^+ = 00100_2 = 4_{10}$$

$$N = 11100_2 = 28_{10}$$

$J_0 = 1$	$K_0 = 1$	$Q_0^+ = 1$
$J_1 = 0$	$K_1 = 1$	$Q_1^+ = 0$
$J_2 = 0$	$K_2 = 0$	$Q_2^+ = 1$
$J_3 = 0$	$K_3 = 0$	$Q_3^+ = 0$
$J_4 = 0$	$K_4 = 0$	$Q_4^+ = 0$

$$N^+ = 11101_2 = 29_{10}$$

$$N = 11101_2 = 29_{10}$$

$J_0 = 1$	$K_0 = 1$	$Q_0^+ = 0$
$J_1 = 1$	$K_1 = 1$	$Q_1^+ = 1$
$J_2 = 0$	$K_2 = 0$	$Q_2^+ = 1$

$J_3=0$	$K_3=0$	$Q_3^+=1$
$J_4=0$	$K_4=0$	$Q_4^+=1$
$N^+=11110_2=30_{10}$		
$N=11100_2=30_{10}$		
$J_0=0$	$K_0=1$	$Q_0^+=0$
$J_1=0$	$K_1=1$	$Q_1^+=0$
$J_2=0$	$K_2=0$	$Q_2^+=1$
$J_3=0$	$K_3=1$	$Q_3^+=0$
$J_4=0$	$K_4=1$	$Q_4^+=0$
$N^+=00100_2=4_{10}$		
$N=11111_2=31_{10}$		
$J_0=0$	$K_0=1$	$Q_0^+=0$
$J_1=1$	$K_1=1$	$Q_1^+=0$
$J_2=1$	$K_2=1$	$Q_2^+=0$
$J_3=1$	$K_3=1$	$Q_3^+=0$
$J_4=1$	$K_4=1$	$Q_4^+=0$
$N^+=00100_2=28_{10}$		

Самовідновність пристрою доведена, тому можна зобразити повний перемикальний граф функціонування лічильника (рис. 11.25).

Схему лічильника реалізуємо на ІМС ТТЛ 555 серії. Електрична схема лічильника зображена на рис. 11.26. Функція реверсу забезпечується використанням п'яти логічних елементів Виключне АБО та додаткового входу керування напрямком лічби m , при цьому при $m=0$ забезпечується режим підсумовування, а при $m=1$ – режим віднімання.

При реалізації електричної схеми лічильника використовуються такі мікросхеми:

- К555ТВ6 – два JK-тригера зі скиданням;
- К555ЛА4 – три логічні елементи 3І-НІ;
- К555ЛИ1 – чотири логічні елементи 2І;
- К555ЛЛ1 – чотири логічні елементи 2АБО;
- К555ЛИ6 – два логічні елементи 4І;

КМ555ЛП5 – чотири 2-входових елементи “виключне АБО”.

На схемі лічильника (рис. 11.26) введено такі позначення:

- DD1, DD2, DD3 – К555ТБ6;
- DD4, DD5 – КМ555ЛПІ5;
- DD6 – К555ЛІА4;
- DD7 – К555ЛІІ1;
- DD8 – К555ЛІЛ1;
- DD9 – К555ЛІІ6.

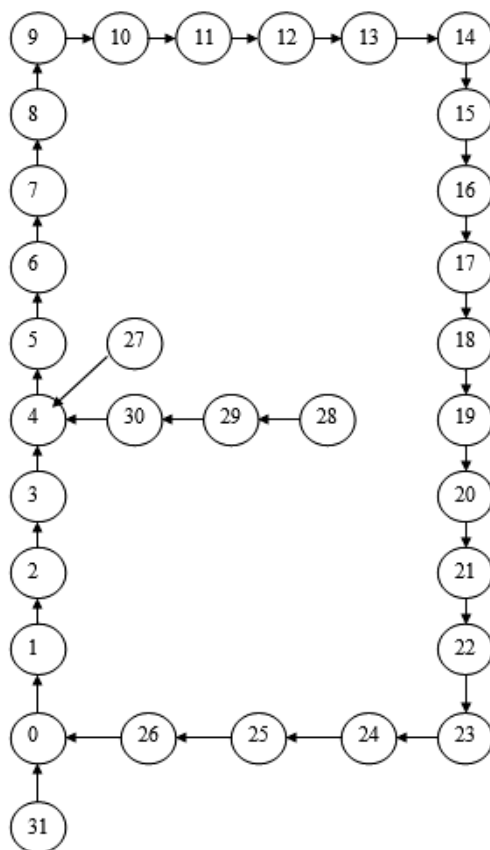
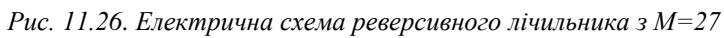


Рис. 11.25. Повний перемикальний граф лічильника

Проведене на ЕОМ моделювання спроектованого лічильника (рис. 11.27 та рис. 11.28) підтверджує вірність роботи пристрою, тому що отримані часові діаграми повністю відповідають теоретичним результатам.



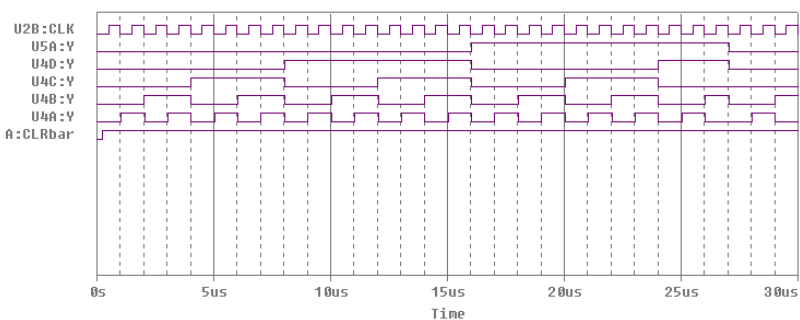


Рис. 11.27. Часові діаграми лічильника при прямій лічбі (режим підсумовування)

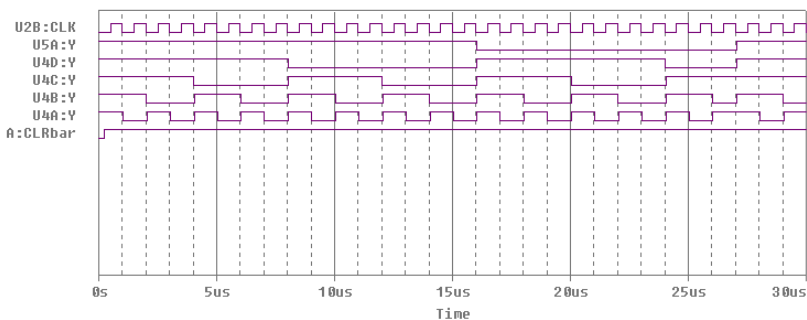


Рис. 11.28. Часові діаграми лічильника при зворотній лічбі (режим віднімання)

11.8.2. Проектування недвійкового програмованого підсумовувального лічильника на ІМС двійкового лічильника

Спроекуємо недвійковий програмований підсумовувальний лічильник на ІМС двійкового лічильника СТ2 з коефіцієнтами перерахунку $M = 5 \div 7$.

Розрядність лічильника $2^{n-1} \leq M_{max} = 7 \leq 2^n \Rightarrow n = 3$.

Кількість надлишкових станів:

$$M_{H1} = 2^n - M_1 = 2^3 - 5 = 3;$$

$$M_{H2} = 2^n - M_2 = 2^3 - 6 = 2;$$

$$M_{H3} = 2^n - M_3 = 2^3 - 7 = 1.$$

Для програмування лічильника необхідно дворозрядний сигнал керування:

$$m_1 = 0 \quad m_0 = 0 \quad \Rightarrow \quad M_1 = 5$$

$$m_1 = 0 \quad m_0 = 1 \quad \Rightarrow \quad M_2 = 6$$

$$m_1 = 1 \quad m_0 = 0 \quad \Rightarrow \quad M_3 = 7$$

Для закінчення циклу лічби необхідно визначити сигнал примусового скиду R_m , для визначення якого складаємо таблицю перемикачів лічильника (табл. 11.7).

Таблиця 11.7. Перемикальна таблиця програмованого лічильника ($M=5\dots7$)

N	i	m ₁	m ₀	Q ₂	Q ₁	Q ₀	R _m
0	0	0	0	0	0	0	0
1	1	0	0	0	0	1	0
2	2	0	0	0	1	0	0
3	3	0	0	0	1	1	0
4	4	0	0	1	0	0	0
5	5	0	0	1	0	1	1
6...7	6...7	0	0	1	1	x	x
0	8	0	1	0	0	0	0
1	9	0	1	0	0	1	0
2	10	0	1	0	1	0	0
3	11	0	1	0	1	1	0
4	12	0	1	1	0	0	0
5	13	0	1	1	0	1	0
6	14	0	1	1	1	0	1
7	15	0	1	1	1	1	x
0	16	1	0	0	0	0	0
1	17	1	0	0	0	1	0
2	18	1	0	0	1	0	0
3	19	1	0	0	1	1	0
4	20	1	0	1	0	0	0
5	21	1	0	1	0	1	0
6	22	1	0	1	1	0	0
7	23	1	0	1	1	1	1

За табл. 11.7 визначаємо функцію збудження сигналу скиду R_m .

Q_2	Q_0				Q_0				m_0
	0	1	3	2	18	19	17	16	
	4	5	7	6	22	23	21	20	
	12	13	15	14	30	31	29	28	
	8	9	11	10	26	27	25	24	
Q_1				m_1					

Q_2	Q_0				Q_0				m_0
		1	x	x					
		x	1						
					x	x	x	x	
					x	x	x	x	
Q_1				m_1					

$$R_m = Q_1 Q_2 \overline{m_1} + Q_0 Q_1 Q_2 + Q_0 Q_2 \overline{m_0} \overline{m_1} = Q_1 Q_2 \overline{m_1} \cdot Q_0 Q_1 Q_2 \cdot Q_0 Q_2 \overline{m_0} \overline{m_1}$$

За знайденою функцією збудження сигналу побудуємо схему лічильника (рис. 11.29).

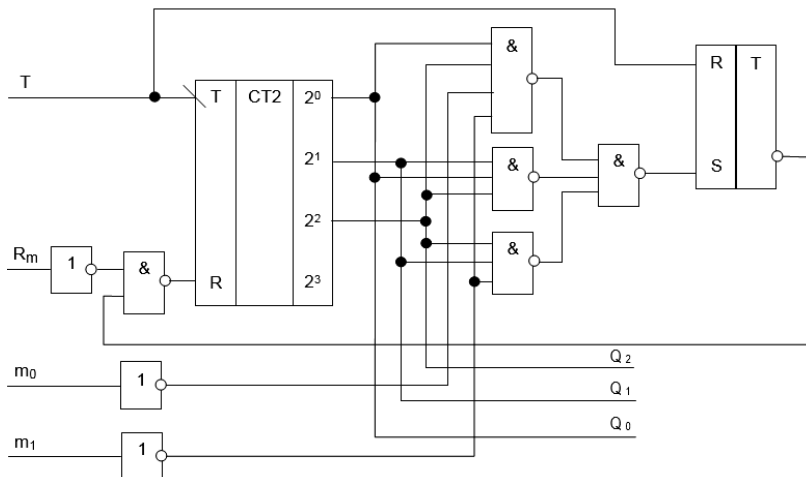


Рис. 11.29. Схема програмованого лічильника з $M = 5 \div 7$

Проаналізуємо самовідновність лічильника.

$$m_1 = m_0 = 0 \Rightarrow R_m = Q_1 Q_2 + Q_0 Q_1 Q_2 + Q_0 Q_2 = Q_1 Q_2 + Q_0 Q_2$$

$$6_{10} = 110_2 \Rightarrow R_m = 1 \Rightarrow 0_{10}$$

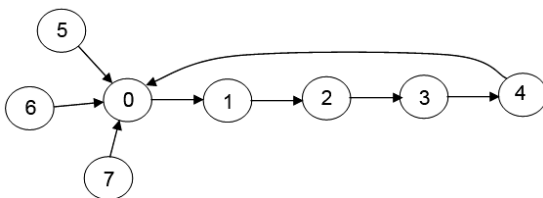
$$7_{10} = 111_2 \Rightarrow R_m = 1 \Rightarrow 0_{10}$$

$$m_1 = 0 \quad m_0 = 1 \Rightarrow R_m = Q_1 Q_2 + Q_0 Q_1 Q_2 = Q_1 Q_2$$

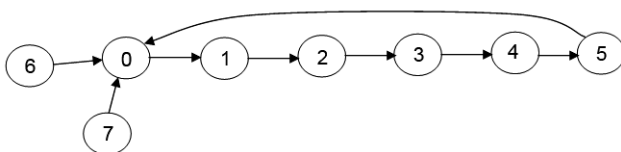
$$7_{10} = 111_2 \Rightarrow R_m = 1 \Rightarrow 0_{10}$$

Проведений аналіз самовідновності довів вірність функціонування пристрою. За аналізом самовідновності побудуємо повний перемикальний граф (рис. 11.30).

$$m_1 = 0 \quad m_0 = 0 \quad \Rightarrow \quad M = 5$$



$$m_1 = 0 \quad m_0 = 1 \quad \Rightarrow \quad M = 6$$



$$m_1 = 1 \quad m_0 = 0 \quad \Rightarrow \quad M = 7$$

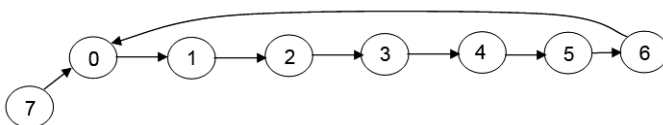


Рис. 11.30. Повний перемикальний граф програмованого лічильника

На рис. 11.31 наведені часові діаграми роботи програмованого лічильника, які пояснюють його роботу.

Для реалізації схеми програмованого лічильника необхідно: двійковий лічильник СТ2, RS-тригер, три інвертори, три тривходових елементи І-НІ, один чотиривходовий елемент І-НІ та один двовходовий елемент І-НІ. Тому для реалізації схеми вибираємо такі ІМС:

- К561ІЕ10 (два чотирирозрядних синхронних двійкових лічильники);
- К561ТР2 (чотири RS-тригера);
- К555ЛА3 (чотири двовходових елементи І-НІ);
- К555ЛА4 (три тривходових елементи І-НІ);
- К555ЛА1 (два чотиривходових елементи І-НІ).

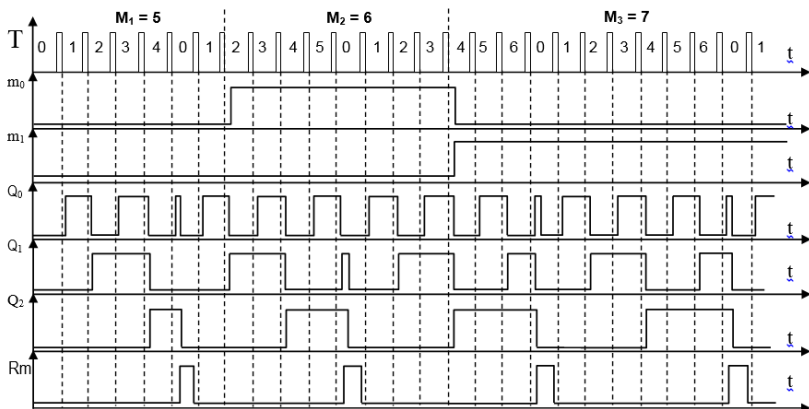


Рис. 11.31. Часові діаграми роботи програмованого лічильника з $M=5\div 7$

Параметри вибраних мікросхем наведені в таблиці 11.8.

Таблиця 11.8. Параметри вибраних мікросхем

Тип ІМС	$I^{\circ}_{\text{вих}}, \text{мА}$	$I^1_{\text{сп}}, \text{мА}$	$I^{\circ}_{\text{сп}}, \text{мА}$	$\bar{t}_{\text{зп}}, \text{нс}$	$U^1_{\text{вх}}, \text{В}$ min, В	$U^{\circ}_{\text{вх}}, \text{В}$ max, В	$U^{\circ}_{\text{вих}}, \text{В}$ max, В	$U^1_{\text{вих}}, \text{В}$ min, В	$I^1_{\text{вих}}, \text{мкА}$ max, мкА
K555ЛA3	8	1,6	4,4	15	2	0,8	2,7	0,5	20
K555ЛA4	8	1,2	3,3	15	2	0,8	2,7	0,5	20
K555ЛA1	8	0,8	2,2	15	2	0,8	2,7	0,5	20
K561IE10	6	0,05	0,05	120	2	0,7	2,8	0,6	40
K561TP2	6	2	2	300	2	0,7	2,8	0,6	40

Отримана в результаті проектування електрична схема спроектованого недвійкового лічильника наведена на рис. 11.32.

Контрольні питання

1. Назвіть основні параметри та класифікаційні ознаки лічильників.

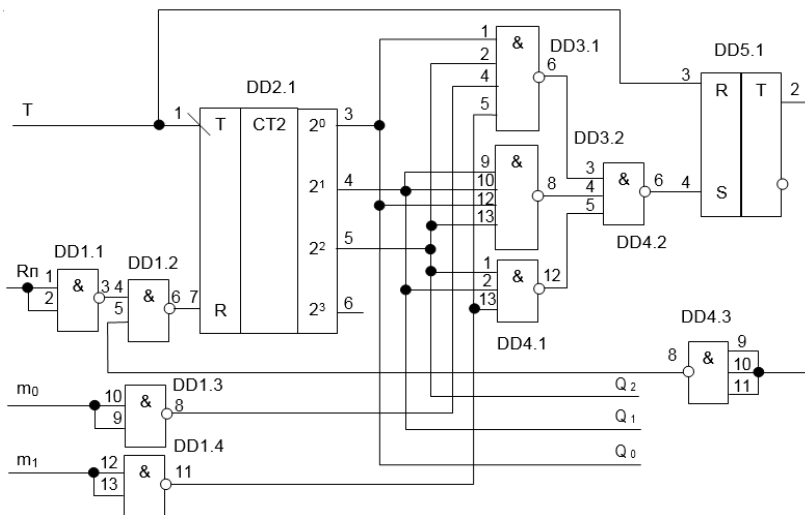


Рис. 11.32. Електрична схема спроектованого лічильника

2. Поясніть принцип роботи реверсивного лічильника.
3. Поясніть принцип дії програмованого лічильника.
4. Якими способами можна побудувати лічильник з довільним коефіцієнтом ділення?
5. Як здійснюється попередня установка лічильників?
6. Поясніть принцип побудови схеми дискретної затримки імпульсних сигналів.
7. Чим відрізняються синхронні та асинхронні лічильники?
8. Як розрахувати коефіцієнт ділення програмованого лічильника, що працює в режимі додавання коду?
9. Як розрахувати коефіцієнт ділення програмованого лічильника, що працює в режимі віднімання коду?
10. Як розрахувати коефіцієнт ділення лічильника зі зворотним зв'язком на основі логічних елементів?
11. Поясніть, яким чином можна побудувати лічильник з довільним коефіцієнтом ділення, використовуючи дешифратори двійкового коду?

12. ГЕНЕРАТОРИ ІМПУЛЬСІВ

12.1. Загальна характеристика генераторів імпульсів

Електронний генератор виконує функції перетворення енергії джерела постійного струму в електричні коливання необхідної форми, частоти та потужності. Електронні генератори є невід'ємною частиною багатьох вимірювальних приладів: частотомірів, осцилографів, вимірювачів часових інтервалів, прийомопередавачів, систем індикації та ін. Основними параметрами генераторів є: форма коливань, частота коливань f , відносна нестійкість частоти коливань $\Delta f/f$, а також корисна і споживана потужності.

Будь-який генератор може працювати в одному з таких режимів:

- очікування;
- автоколивань;
- синхронізації;
- ділення частоти.

В режимі очікування генератор працює з одним стійким станом рівноваги. Зовнішній запускаючий імпульс викликає стрибкоподібний перехід генератора до нового стану, що не є стійким. У цьому стані, який називають тимчасово стійким, генератор знаходиться протягом часу, обумовленого параметрами його схеми. Таким чином, очікувальний генератор генерує тільки один імпульс визначеної тривалості при впливі на нього зовнішнього запускаючого імпульсу. Такий режим роботи генератора використовують для формування часових інтервалів (формування імпульсів за тривалістю), для ділення частоти та для інших цілей (наприклад, для затримки сигналів).

В автоколивальному режимі генератор має два нестійкі стани і жодного стійкого. Генератор у такому режимі без зовнішніх впливів переходить з одного стану до іншого і навпаки.

Генератор є нелінійним пристроєм. Узагальнена схема генератора складається з підсилювача, частотозадавального ко-

ла позитивного зворотного зв'язку (ПЗЗ) і кола негативного зворотного зв'язку (НЗЗ). Коло ПЗЗ визначає умови збудження коливань, частоту і швидкість наростання амплітуди вихідного коливання генератора. Наростання амплітуди коливань буде відбуватися доти, поки дія нелінійного НЗЗ не обмежить її зростання.

При включенні живлення виникають коливання, зумовлені нестационарними процесами: зарядом ємностей, індуктивностей, а також перехідними процесами в активних елементах. Ці коливання, потрапляючи на вхід підсилювача, підсилюються останнім і через коло ПЗЗ знову надходять на вхід підсилювача.

Для виникнення коливань необхідні дві умови, які прийнято називати умовами балансу амплітуд і фаз:

$$\begin{cases} |K\beta| = 1; \\ \arg(K\beta) = \varphi_k + \varphi_\beta = 0, \end{cases}$$

де K – коефіцієнт підсилення підсилювача; β – коефіцієнт передачі кола ПЗЗ; φ_k – фазовий зсув коливань, внесений підсилювачем; φ_β – фазовий зсув коливань, внесений колом ПЗЗ.

Для формування коливань потрібної частоти використовують кола, що містять реактивні елементи – ємності або індуктивності. Якщо в генераторах використовують тільки RC-кола, то такі генератори мають назву мультивібраторів.

Мультивібратори використовуються для отримання імпульсів прямокутної форми. Генератор містить як мінімум один реактивний елемент, наприклад, конденсатор, ємність якого разом з активним опором визначає тривалість вихідних імпульсів.

Для проведення вимірювань найбільш важливою є відносна нестабільність частоти коливань $\Delta f/f$. За цим параметром генератори умовно можна розділити на три групи:

- низькостабільні $10^{-3} > \Delta f/f > 10^{-6}$;
- середньостабільні $10^{-6} > \Delta f/f > 10^{-9}$;
- високостабільні $\Delta f/f < 10^{-9}$.

Детальніше з характеристиками стабільних генераторів ми познайомимся у розділі, присвяченому стабілізації частоти генерованих коливань.

12.2. Очікувальні мультивібратори (одновібратори)

Мультивібратори, що працюють в очікувальному режимі, отримали назву одновібраторів (ОВ). Розглянемо принципи побудови таких генераторів на різних елементах цифрової техніки.

12.2.1. Одновібратор на RS-тригері

Схема одновібратора, побудованого з використанням RS-тригера, наведена на рис. 12.1.

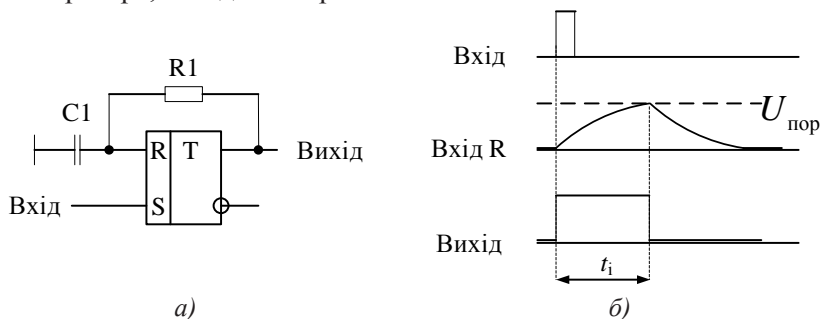


Рис. 12.1. Схема одновібратора, побудованого на RS-тригері (а)
та часові діаграми його роботи (б)

Схема працює таким чином. Після надходження на вхід S тригера імпульсу запуску на виході схеми встановлюється рівень лог. 1 і починається заряд конденсатора C1 через резистор R1. Як тільки напруга на конденсаторі сягне порогового рівня, тригер переключиться до стану лог. 0, а схема перейде до початкового стану. Тривалість імпульсу на виході одновібратора буде зумовлена часом заряду конденсатора C1 до порогового рівня.

Знайдемо тривалість імпульсу такого одновібратора. Відомо, що напруга на конденсаторі описується співвідношенням

$$U_c = A(1 - e^{-t/\tau}), \quad (12.1)$$

де A – амплітуда вхідного сигналу, τ – стала часу RC-кола.

Для розрахунку тривалості імпульсу приймаємо $U_c = U_{\text{пор}}$, $t = t_i$, $A = U_{\text{вих}}$. Перепишемо (12.1) з врахуванням прийнятих позначень:

$$U_{\text{пор}} = U_{\text{вих}} (1 - e^{-t_i/\tau}).$$

Розв'язавши рівняння відносно t_i , отримаємо:

$$t_i = \tau \ln \frac{U_{\text{вих}}}{U_{\text{вих}} - U_{\text{пор}}}.$$

Для мікросхем, виготовлених за технологією КМОН, з урахуванням того, що $U_{\text{вих}} = U_{\text{ж}}$, а $U_{\text{пор}} = U_{\text{ж}}/2$, останнє співвідношення спрощується і набуває такого вигляду:

$$t_i = \tau \ln \frac{U_{\text{ж}}}{U_{\text{ж}} - U_{\text{ж}}/2} = \tau \ln 2 = 0,693\tau. \quad (12.2)$$

Однак у такого одновібратора є один суттєвий недолік. Оскільки час заряду та розряду конденсатора приблизно однаковий, то період слідування вхідних імпульсів не може бути меншим, ніж $2t_i$. Це пов'язано з тим, що коли конденсатор C1 не повністю розрядиться до моменту надходження наступного вхідного імпульсу, то час його заряду до порогового рівня буде меншим, ніж при заряді від нульового рівня, тобто тривалість вихідного імпульсу ОВ зменшиться, що недопустимо.

Для зменшення часу розряду використовують пришвидшувальне коло, що складається з діода VD1 та резистора R2 (рис. 12.2). Опір резистора R2 вибирається набагато меншим, ніж опір R1 ($R_2 \ll R_1$). При заряді конденсатора C1 діод закритий і наявність резистора R2 не впливає на швидкість заряду, а при розряді діод відкривається і струм розряду протікає паралельно по двох колах. А оскільки $R_2 \ll R_1$, то струм через R2 протікає набагато більший, ніж через R1, тобто час, потрібний для розряду конденсатора C1 значно скорочується.

12.2.2. Одновібратор на RS-тригері з інверсними входами

Оскільки більшість мікросхем містять RS-тригери з інверсними входами, розглянемо одновібратор, побудований на такому тригері. Використаємо вхід R для запуску одновібратора

(для запуску можна використовувати будь-який вхід). Схема такого ОВ наведена на рис. 12.3.

На відміну від ОВ, побудованого на RS-тригері з прямими входами, після подачі на вхід імпульсу запуску починається розряд конденсатора C1. Тривалість імпульсу буде дорівнювати часу розряду конденсатора від максимальної напруги на виході Q до порогового рівня, що ілюструють часові діаграми роботи ОВ (рис. 12.3, б).

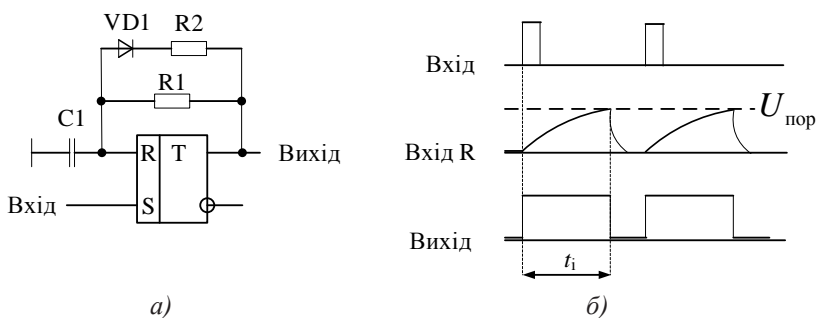


Рис. 12.2. Схема одновібратора, побудованого на RS-тригері зі скороченим часом відновлення (а) та часові діаграми його роботи (б)

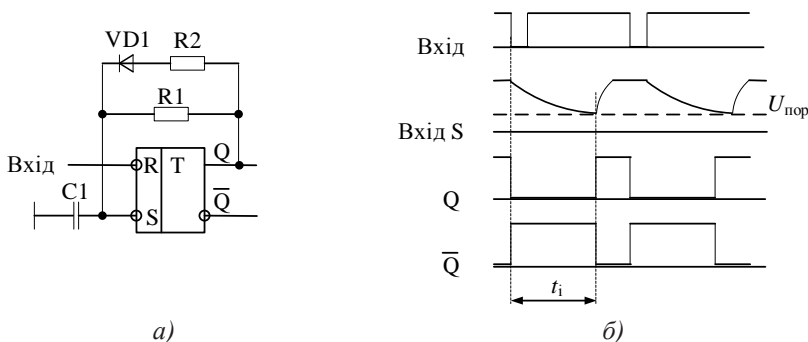


Рис. 12.3. Схема одновібратора, побудованого на RS-тригері з інверсними входами (а) та часові діаграми його роботи (б)

Для обох розглянутих одновібраторів необхідно завжди виконувати умову, яка полягає в тому, що тривалість вхідного імпульсу має бути завжди менша, ніж тривалість вихідного. Якщо $t_{вх} > t_i$, то порушується умова нормальної роботи тригера (на входах тригера встановлюється заборонений стан). Для RS-три-

гера заборонений стан $S=R=1$, а для RS-тригера з інверсними входами заборонений стан $S=R=0$.

Для того, щоб запобігти такій ситуації, на вході ОВ потрібно встановити формувач коротких імпульсів (рис. 12.4 і 12.5).

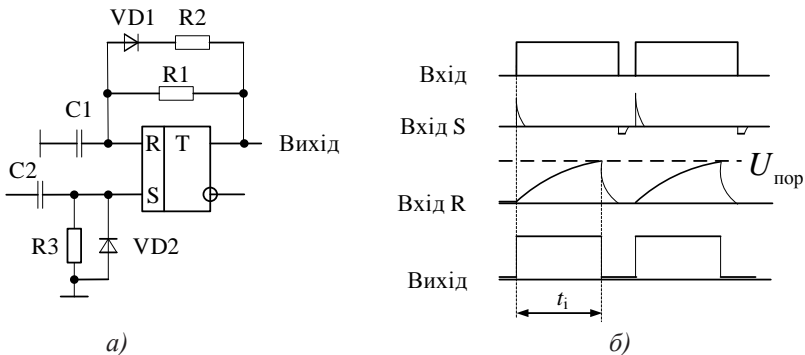


Рис. 12.4. Схема одновібратора, побудованого на RS-тригері з формувачем коротких імпульсів на вході (а) та часові діаграми його роботи (б)

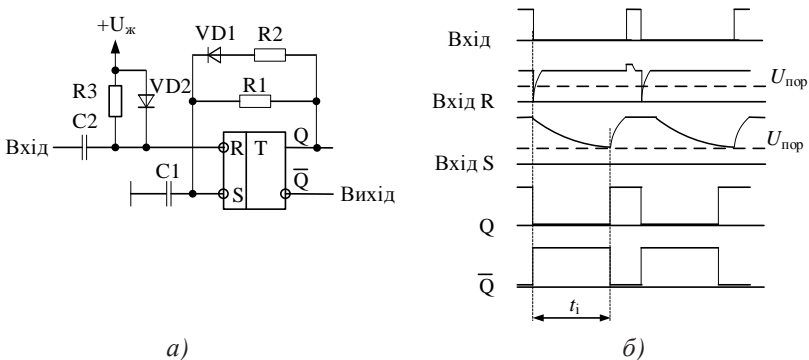


Рис. 12.5. Схема одновібратора, побудованого на RS-тригері з інверсними входами і формувачем коротких імпульсів на вході (а) та часові діаграми його роботи (б)

Формувачі коротких імпульсів побудовані з використання диференціальних кіл, а діод VD2 призначений для обмеження напруги на входах тригерів у безпечних межах.

12.2.3. Одновібратор, не чутливий до тривалості вхідного імпульсу

Мультивібратор, не чутливий до тривалості вхідного імпульсу, можна побудувати, використовуючи RSD- або RSJK-тригер. На рис. 12.6 наведена схема мультивібратора, виконаного на RSD-тригері. Оскільки вхід С використовується для запуску одновібратора, то спрацювати ОВ буде за фронтом вхідного сигналу.

Після подачі вхідного імпульсу у D-тригер записується лог. 1, оскільки на вхід D подається напруга живлення. Після перемикання тригера до стану лог. 1 починається заряд конденсатора С1. Тривалість вихідного імпульсу буде дорівнювати часу заряду конденсатора до порогового рівня. Як тільки напруга на вході R досягне $U_{\text{пор}}$, тригер перемикається до стану лог. 0 і конденсатор швидко розряджається через коло R2, VD1. Часові діаграми роботи такого одновібратора наведені на рис. 12.6, б.

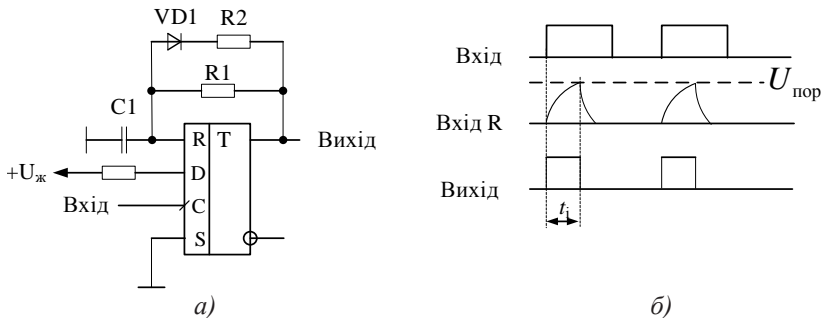


Рис. 12.6. Схема одновібратора, не чутливого до тривалості вхідного імпульсу (а), та часові діаграми його роботи (б)

12.2.4. Одновібратори на логічних елементах

Для побудови одновібраторів можна використовувати логічні елементи та інтегрувальні або диференціювальні кола. На рис. 12.7 наведена схема одновібратора, в якому використовується диференціювальне коло.

На вхід такого ОВ потрібно подавати низький рівень вхідної напруги для його запуску (рис. 12.7, б). До подачі вхідного сигналу у точці А напруга дорівнює лог. 0, а отже на виході інвертора формується рівень лог. 1 і, як наслідок, на виході Q формується рівень лог. 0. При подачі на вхід елементу І-НІ нульового рівня на його виході Q формується рівень лог. 1 і цей

стрибок напруги через диференціувальне коло C1, R1 передається на вхід інвертора до точки А. На виході \bar{Q} буде сформовано лог. 0, який буде утримувати логічний елемент І-НІ у стані $Q=1$ до закінчення формування вихідного імпульсу.

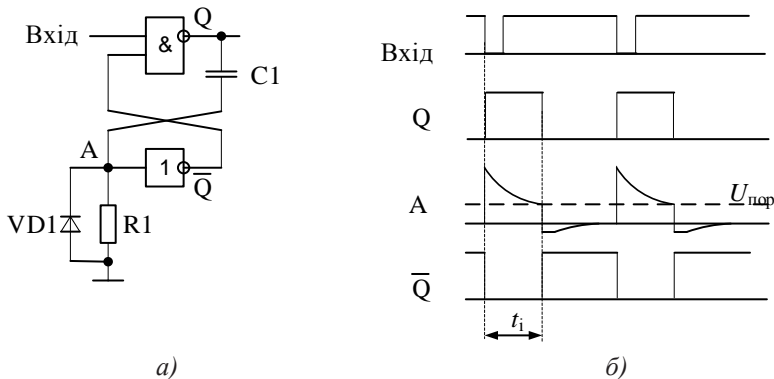


Рис. 12.7. Схема одновібратора на логічних елементах (а) та часові діаграми його роботи (б)

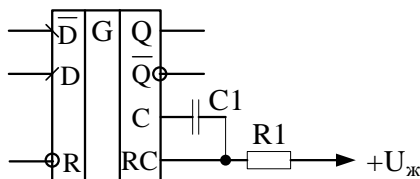
Далі починається заряд конденсатора C1 і, як наслідок, напруга у точці А зменшується. Коли вона сягне порогового значення, інвертор сформує на виході \bar{Q} рівень лог. 1 і на виході Q встановиться лог. 0 (при цьому формування імпульсу закінчується). Перепад напруги з лог. 1 в лог. 0 з виходу елемента І-НІ передається до точки А. Для запобігання ушкодження входних кіл логічного елемента призначений діод VD1 (як правило, вбудований у мікросхему). Він обмежує від'ємну напругу на вході А до величини -0,7 В.

Аналогічно можна побудувати аналог RS-тригера на логічних елементах і реалізувати ОВ з використанням інтегрувального кола.

12.2.5. Одновібратори в інтегральному виконанні

В багатьох серіях мікросхем випускаються інтегральні мультівібратори, які є універсальними і дозволяють здійснювати їх запуск як фронтом, так і спадом вхідного сигналу, а також встановлювати їх до початкового стану в будь-який момент ча-

КР15335АГЗ



Стан входів			Стан виходів	
R	\overline{D}	D	Q	\overline{Q}
0	X	X	0	1
X	1	X	0	1
X	X	0	0	1
1	0	$0 \rightarrow 1$	\sqcap	\sqcup
1	$1 \rightarrow 0$	1	\sqcap	\sqcup
$0 \rightarrow 1$	0	1	\sqcap	\sqcup

a)

6)

З таблиці функціонування видно, що ОВ може запускатися як за фронтом (входи D і R), так і за спадом (вхід \overline{D}).

Залишається ще пояснити, що таке повторний запуск. Якщо після запуску ОВ під час формування імпульсу в момент часу t_1 (імпульс на виході ще не закінчився) на вхід запуску знову подати імпульс, то на виході, починаючи з цього моменту часу, буде продовжуватися формування вихідного імпульсу. При цьому тривалість вихідного імпульсу в такій ситуації буде складати величину $t_{\text{вих}} = t_1 + t_i$, де t_i – тривалість імпульсу на виході одновібратора при його запуску одиночним імпульсом, тобто тривалість зумовлена сталою часу RC-кола, підключеного до входів мікросхеми (рис. 12.9).

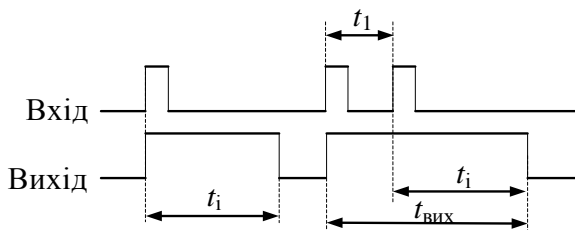


Рис. 12.9. Часові діаграми роботи ОВ з повторним запуском

Таким чином, можна подавати не тільки 2, а довільну кількість імпульсів для продовження формування імпульсу на виході одновібратора.

12.3. Мультивібратори

12.3.1. Мультивібратори на RS-тригерах

Мультивібратори на RS-тригерах можна побудувати на основі схем, наведених у п. 12.2.1 та 12.2.2. Для цього потрібно, щоб сигнал з виходу одного одновібратора приводив до запуску другого одновібратора, для побудови яких використовуються входи S і R. На рис. 12.10 наведена схема мультивібратора, побудованого на RS-тригері з прямими входами, та часові діаграми сигналів у різних точках схеми.

Період слідування вихідного сигналу дорівнює $T_{\text{вих}} = t_{i1} + t_{i2}$, а тривалість імпульсів з урахуванням (12.2) становить

$$t_{i1} = R_3 C_2 \ln \frac{U_{\text{ж}}}{U_{\text{ж}} - U_{\text{ж}} / 2} = R_3 C_2 \ln 2 = 0,693 R_3 C_2,$$

$$t_{i2} = R_1 C_1 \ln \frac{U_{\text{ж}}}{U_{\text{ж}} - U_{\text{ж}} / 2} = R_1 C_1 \ln 2 = 0,693 R_1 C_1.$$

Розглянемо роботу мультивібратора детально. Починати аналіз роботи можна з будь-якого стану на виході. Припустимо, що на виході Q встановлений рівень лог. 1. Тоді конденсатор C1 заряджається і коли напруга на ньому досягне порогового рівня, тригер переключиться до стану лог. 0. В результаті цього на виході \bar{Q} встановиться рівень лог. 1 і почне заряджатися конденсатор C2. Оскільки на виході Q встановлюється рівень лог. 0, то

конденсатор $C1$ швидко розряджається через пришвидшуюче коло $VD1, R2$. У той момент часу, коли напруга на конденсаторі $C2$ сягне порогового значення, тригер переключиться до стану лог. 1 і знову почнеться заряд конденсатора $C1$ та швидкий розряд конденсатора $C2$ через пришвидшуюче коло $VD2, R4$. Далі весь процес повторюється.

Аналогічно будується мультивібратор на основі RS-тригера з інверсними входами, схема якого наведена на рис. 12.11.

Принцип дії такого мультивібратора аналогічний принципу дії одновібратора, виконаного на такому самому тригері (див. п. 12.2.2).

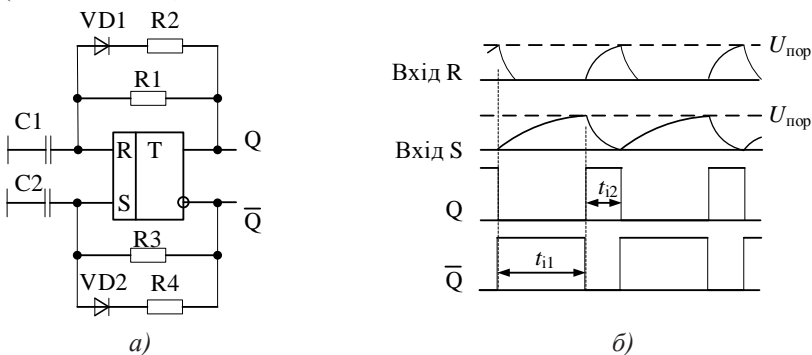


Рис. 12.10. Схема мультивібратора на RS-тригері (а) та часові діаграми його роботи (б)

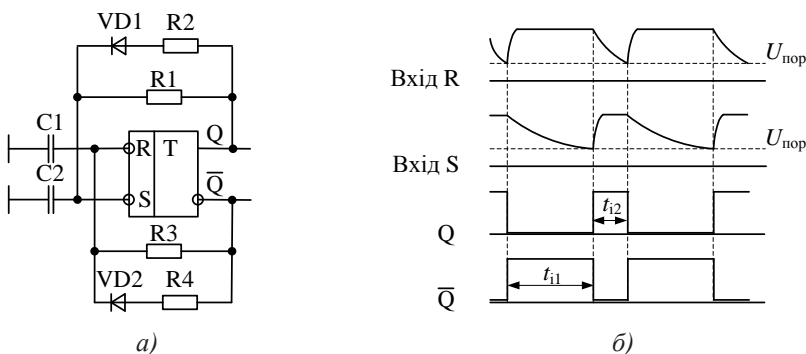


Рис. 12.11. Схема мультивібратора на RS-тригері з інверсними входами (а) та часові діаграми його роботи (б)

12.3.2. Мультивібратори на логічних елементах

При побудові мультивібраторів на логічних елементах використовується велика кількість схемних рішень. Розглянемо деякі з них.

На базі одновібратора на логічних елементах можна побудувати мультивібратор, схема якого наведена на рис. 12.12.

Тривалість імпульсів кожного плеча мультивібратора залежить від часу, за який конденсатори будуть заряджатися до напруги $U_C = U_{\text{Ж}} - U_{\text{пор}}$. Якщо вибрати номінальні значення елементів таким чином, щоб $R_1 = R_2$, $C_1 = C_2$, то такий мультивібратор називається симетричним, а тривалість імпульсів кожного плеча однакова: $t_{i1} = t_{i2}$. Період слідування вихідного сигналу може бути розрахований за формулою

$$T_{\text{вих}} = 2RC \ln \left(\frac{U_{\text{вих}}}{U_{\text{пор}}} \right),$$

де $U_{\text{вих}}$ – амплітуда сигналу на виході інверторів.

Для КМОН-елементів $U_{\text{вих}} = U_{\text{Ж}}$.

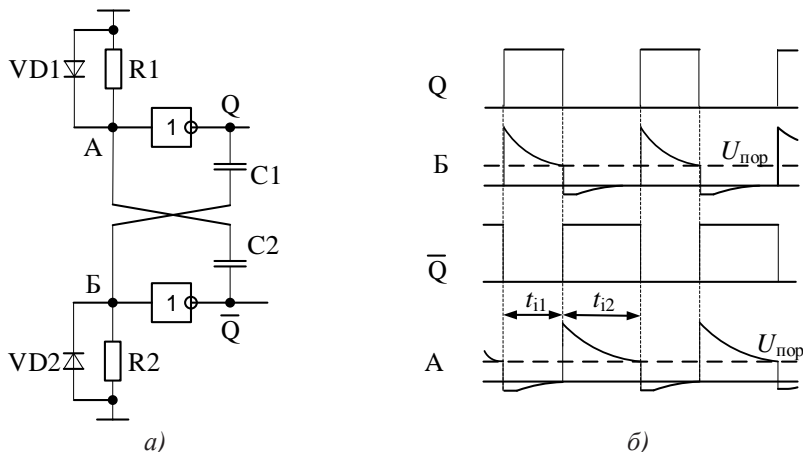


Рис. 12.12. Схема мультивібратора на логічних елементах (а) та часові діаграми його роботи (б)

Використовуючи логічні елементи КМОН, можна побудувати мультивібратор, що має тільки одне коло, яке задає пері-

од слідування вихідних імпульсів. Схема такого мультивібратора наведена на рис. 12.13.

Розглянемо роботу схеми. Припустимо, що напруга в точці Б стрибком змінилася з рівня лог. 0 до рівня лог. 1 (рис. 12.13, б). Тоді у точці А напруга буде дорівнювати рівню лог. 0, а на виході диференціувального кола, що складається з резистора R2 та конденсатора C1, буде сформований стрибок напруги від рівня $U_{\text{пор}}$ до $U_{\text{пор}} + U_{\text{ж}}$. Починається заряд конденсатора по колу вихід DD2 → C1 → R2 → вихід DD1. Як наслідок, напруга в точці Г починає зменшуватись, і коли вона впаде до порогового рівня, інвертор DD1 переключиться з стану лог. 0 до стану лог. 1, оскільки його вхід через резистор R1 підключений до точки Г.

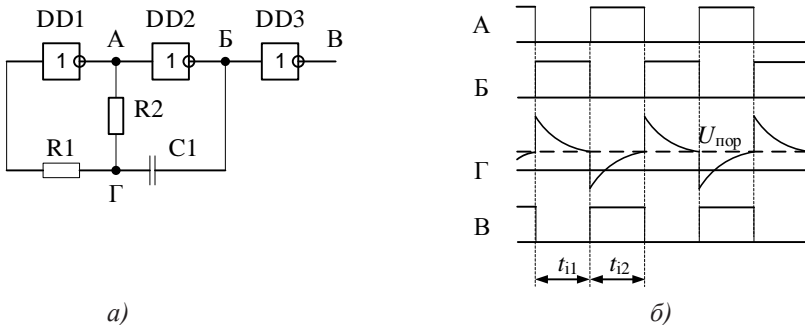


Рис. 12.13. Схема мультивібратора на логічних елементах КМОН (а) та часові діаграми його роботи (б)

Після цього на виході Б напруга стрибком переходить з рівня лог. 1 до рівня лог. 0 і цей стрибок через коло C1, R2 передається в точку Г. Починається перезаряд конденсатора C1 і напруга у точці Г починає збільшуватись. Як тільки напруга у цій точці зростає до порогового рівня, інвертор DD1 переключається до стану лог. 0, на виході DD2 формується рівень лог. 1 і весь процес повторюється.

Інвертор DD3 призначений для запобігання впливу опору навантаження, що підключається на виході генератора, на частоту вихідного коливання. Резистор R1 призначений для запобігання розряду конденсатора C1 через захисні діоди, що знаходяться всередині інвертора, а також для запобігання розряду конденсатора C1 через захисні діоди, встановлені на вході ін-

вертора (всередині мікросхеми) DD1. Для стабільної роботи мультівібратора потрібно виконання умови $R1 > R2$.

Для схеми, наведеної на рис. 12.13,

$$t_{i1} \approx t_{i2},$$

де

$$t_{i1} = R_2 C_1 \ln \frac{U_{\text{ж}} + U_{\text{пор}}}{U_{\text{пор}}}, \quad t_{i2} = R_2 C_1 \ln \frac{2U_{\text{ж}} - U_{\text{пор}}}{U_{\text{ж}} - U_{\text{пор}}}.$$

Якщо $U_{\text{пор}} = U_{\text{ж}}/2$, то період слідування вихідних імпульсів буде дорівнювати $T_{\text{вих}} = t_{i1} + t_{i2} \approx 2,2 R_2 C_1$.

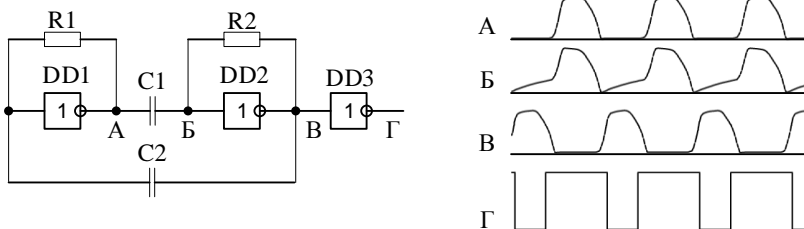
Якщо виключити резистор $R1$ з схеми і з'єднати точку Γ з входом DD1, то буде виконуватись умова

$$t_{i1} \approx t_{i2} = R_2 C_1 \ln \frac{U_{\text{ж}}}{U_{\text{ж}} - U_{\text{ж}}/2} = 0,693 R_2 C_2,$$

а період слідування вихідних імпульсів буде $T_{\text{вих}} \approx 1,4 R_2 C_1$.

У всіх розглянутих раніше схемах генераторів логічні елементи працюють у ключовому режимі. Однак їх можна використати і в лінійному чи близькому до лінійного режимах. Схема такого генератора наведена на рис. 12.14.

Генератор фактично складається з двох інвертувальних підсилювачів. Резистори $R1$ та $R2$ встановлюють робочу точку інверторів на лінійній ділянці амплітудної характеристики. Конденсатор $C2$ формує позитивний зворотний зв'язок, тому вихідний сигнал підсилювача повертається на вхід (завдяки двом каскадам підсилювача зсув фаз між входом та виходом дорівнює 2π , що забезпечує баланс фаз генератора). Логічний елемент DD3 є буферним, тобто він призначений для покращення форми вихідного сигналу. Часові діаграми роботи генератора наведені на рис. 12.14, б.

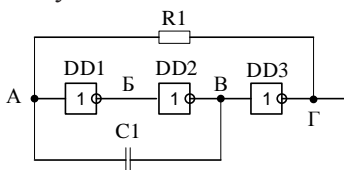


а)

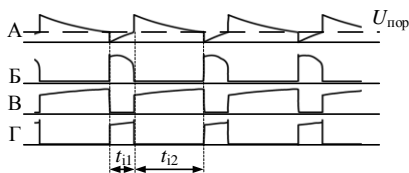
б)

Рис. 12.14. Схема мультивібратора на логічних елементах ТТЛ, що працюють у лінійному режимі (а), та часові діаграми його роботи (б)

Ще один приклад побудови генератора на логічних елементах, що працюють в лінійному режимі, наведено на рис. 12.15, а. Схему зібрано на трьох елементах І-НІ. Часові діаграми роботи генератора показані на рис. 12.15, б. Принцип роботи генератора полягає в наступному. Припустимо, що в початковий момент часу на виході елементу DD3 встановлений рівень лог. 0. Це означає, що на виході елементу DD2 присутній рівень лог. 1. Отже, конденсатор С1 буде заряджатись і на вході елементу сформується стрибок напруги, який переведе елемент DD1 до стану лог. 0.



а)



б)

Рис. 12.15. Схема генератора на трьох елементах І-НІ (а) та часові діаграми його роботи (б)

Заряд конденсатора продовжується доти, поки напруга на вході DD1 не буде дорівнювати $U_{\text{пор}}$. В цей час на правій обкладинці конденсатора позитивний потенціал, а на лівій – негативний (сам конденсатор заряджений до напруги $U_1 - U_{\text{пор}}$, де U_1 – вихідна напруга на виході логічного елементу). Для КМОН елементів $U_1 = U_{\text{ж}}$, а для ТТЛ залежить від опору навантаження і серії мікросхем. Регламентується тільки мінімальний рівень лог. 1 не менше, ніж 2,4 В.

Коли на вході DD1 сформується рівень лог. 0 (відповідно на виході DD2 – лог. 0, а на виході DD3 – лог. 1), то конденсатор почне перезаряджатись від напруги $U_1 - U_{\text{пор}}$ до $U_{\text{пор}}$. В кінці перехідного процесу ліва обкладинка буде заряджена позитивно, а права – негативно. Коли на вході DD1 знову з'явиться рівень лог. 1, то в точці Б сформується лог. 0, а в точці В – лог. 1, тобто конденсатор знову перезаряджається. Але в точці А сформується

ся стрибок напруги, який дорівнює $U_1 + U_{\text{пор.}}$ (оскільки на конденсаторі залишилась напруга від попереднього процесу).

Обчислимо період слідування сигналу на виході генератора. Він складається з двох часових інтервалів. В момент, коли на виході генератора (точка Г) рівень лог. 1, потенціал точки А визначається напругою на конденсаторі. Оскільки умови на конденсаторі не є нульовими, то знайдемо функцію зміни напруги на конденсаторі. Загальний розв'язок диференційного рівняння для напруги на конденсаторі

$$U_C(t) = U_1 + Ae^{-t/R_1C_1}.$$

В момент часу $t = 0$, $U_C(t) = -(U_1 - U_{\text{пор.}})$. Отже, константа $A = -(2U_1 - U_{\text{пор.}})$.

Тривалість імпульсу на виході генератора визначається за умови $U_C(t) = U_{\text{пор.}}$:

$$U_{\text{пор.}} = U_1 - (2U_1 - U_{\text{пор.}})e^{-t/R_1C_1},$$

звідки

$$t_{i1} = R_1C_1 \ln \left(\frac{2U_1 - U_{\text{пор.}}}{U_1 - U_{\text{пор.}}} \right).$$

В ті моменти часу, коли на виході генератора формується рівень лог. 0, потенціал точки А визначається напругою на резисторі:

$$U_R(t) = (U_1 + U_{\text{пор.}})e^{-t/R_1C_1},$$

звідки

$$t_{i2} = R_1C_1 \ln \left(\frac{U_1}{U_1 + U_{\text{пор.}}} \right).$$

Період сигналу дорівнює $T_{\text{вих}} = t_{i1} + t_{i2}$.

12.3.3. Мультивібратори на основі інтегральних одновібраторів

Використовуючи мікросхему КР1533АГЗ, що містить два одновібратора, можна побудувати мультивібратор, схема і часові діаграми якого наведені на рис. 12.16.

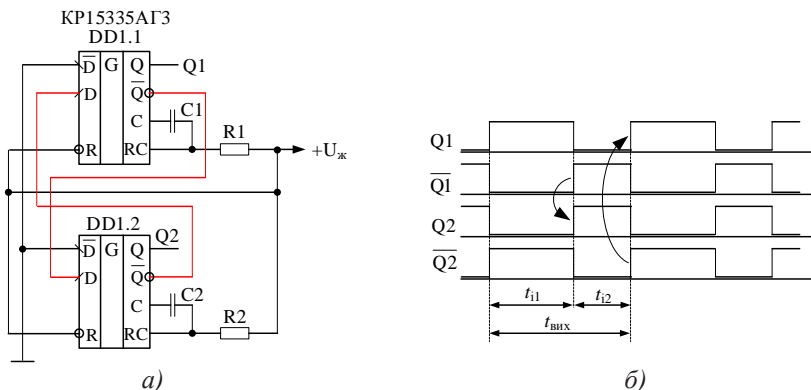


Рис. 12.16. Схема мультивібратора на мікросхемі інтегрального одновібратора КР1533АГЗ (а) та часові діаграми його роботи (б)

Принцип побудови такого генератора дуже простий. Вихідний сигнал верхнього одновібратора запускає формування імпульсу другим одновібратором і навпаки (рис. 12.16, б).

12.3.4. Мультивібратори на основі інтегрального таймера NE555

Функціональна схема інтегрального таймера NE555 наведена на рис. 12.17. Функціональні аналоги, що випускаються в Україні, 1006ВІ1 та 1087ВІ2.

Основні технічні характеристики мікросхеми NE555:

- напруга живлення – (4,5...18) В;
- максимальний вихідний струм – 200 мА;
- споживаний струм – до 206 мА.

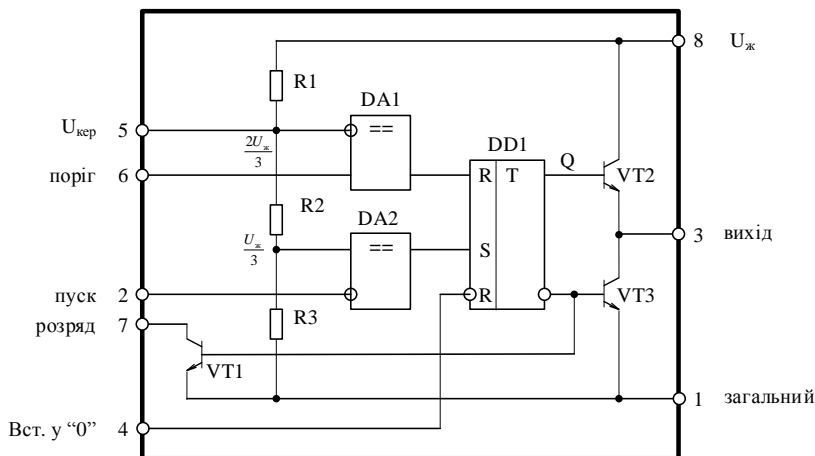


Рис. 12.17. Функціональна схема інтегрального таймера NE555

Серед функціональних блоків виділяють RS-тригер (DD1), компаратори (DA1 і DA2), вихідний підсилювальний каскад, побудований за двотактною схемою, і додатковий транзистор VT1. Призначення транзистора VT1 – розряд частотнозадавального конденсатора при використанні мікросхеми в якості генератора. Встановлення тригера до стану лог. 0 (скидання) відбувається при подачі лог. 1 (рівень від $U_{ж}/2$ до $U_{ж}$) на вхід R. Якщо тригер встановлений до стану лог. 0, то на виході мікросхеми (вив. 3) формується низький рівень напруги (транзистор VT3 відкритий).

За функціональною схемою мікросхеми важко зрозуміти її унікальність. Вся оригінальність полягає у способі керування тригером, а саме формування сигналів керування. Сигнали керування формуються на виходах компараторів DA1 і DA2, на один з входів яких подана опорна напруга. Щоб сформувати сигнали керування, необхідно отримати на входах тригера (виходи компараторів) сигнали високого рівня.

Для запуску таймера необхідно подати на вхід "Пуск" (вив. 2) напругу в межах $0 \dots U_{ж}/3$. Цей сигнал приводить до спрацювання тригера, і на його виході формується сигнал рівня лог. 1. Сигнал, більший ніж $U_{ж}/3$, не викличе будь-яких змін у стані мікросхеми, тому що опорна напруга для компара-

тора DA2 становить $U_{\text{ж}}/3$.

Зупинка таймера відбувається після скидання тригера. Для цього напруга на вході $U_{\text{пор}}$ (вив. 6) має перевищити $2U_{\text{ж}}/3$ (опорна напруга для компаратора DA1 становить $2U_{\text{ж}}/3$). При скиданні тригера на виході мікросхеми встановлюється сигнал низького рівня і відбувається розряд частотнозадавального конденсатора.

Регулювання опорної напруги може проводитися шляхом підключення додаткового резистора або джерела живлення до виводу 5 мікросхеми.

На рис. 12.18 показана схема одновібратора, побудованого з використанням таймера NE555. Пусковий імпульс подається на вхід нижнього компаратора (вив. 2), інверсний вхід R (вхід скидання) тригера (вив. 4) не використовується, а на нього подається напруга живлення. Інверсний вхід верхнього компаратора (вив. 5) для уникнення впливу завад заземлюється за змінним струмом через конденсатор фільтра C1. Транзистор VT1 виконує функцію розрядного ключа (рис. 12.17) і підключається паралельно частотнозадавальному конденсатору C2.

Принцип дії одновібратора ілюструється часовими діаграмами, наведеними на рис. 12.19.

У початковому стані на виході таймера встановлений рівень лог. 0, транзистор VT1 (рис. 12.17) відкритий, а напруга на конденсаторі C2 дорівнює нулю.

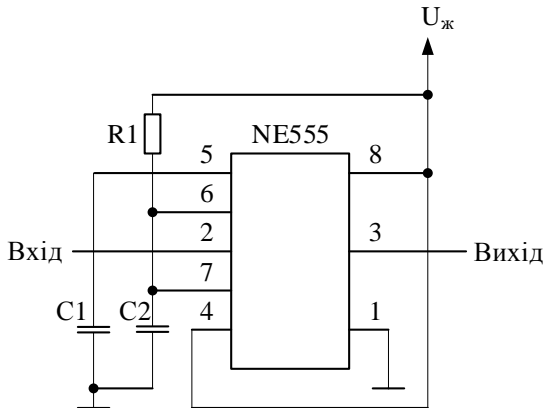


Рис. 12.18. Схема одновібратора на таймері NE555

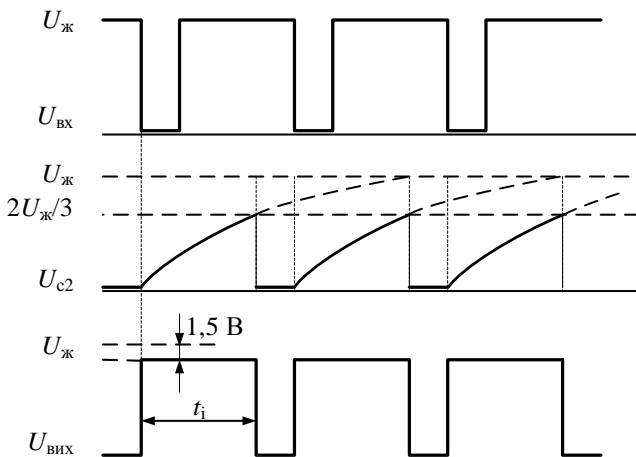


Рис. 12.19. Часові діаграми роботи одновібратора на базі таймера NE555

Після приходу негативного фронту пускового імпульсу $U_{вх}$ компаратор DA2 переключається в лог. 1 і встановлює тригер DD1 до стану лог. 1. На виході таймера встановлюється високий рівень ($U_{ж} - 1,5 \text{ В}$), а транзистор VT1 закривається і починається заряд конденсатора C2 через резистор R1 від джерела живлення.

Як тільки напруга на конденсаторі C2 сягне значення $2U_{ж}/3$, компаратор DA1 переключається до стану лог. 1 і, як наслідок, тригер переключається до стану лог. 0, а вихідна напруга таймера стає близькою до нуля. Транзистор VT1 (вив. 7) відкривається і конденсатор C2 швидко розряджається.

Якщо наступний вхідний імпульс надійде раніше, ніж закінчиться формування вихідного імпульсу, то одновібратор на нього не відреагує.

Тривалість вихідного імпульсу можна знайти з рівняння для напруги на конденсаторі:

$$U_c = U_{ж}(1 - e^{-t/\tau}),$$

де $\tau = R_1 C_2$ – стала часу RC-кола.

Оскільки заряд конденсатора закінчується, коли напруга

на ньому стає рівною $2U_{ж}/3$, то можна скласти рівняння

$$U_{ж}(1 - e^{-t/\tau}) = \frac{2U_{ж}}{3},$$

звідки

$$t_i = \tau \ln 3 = 1,1R_1C_2.$$

Використовуючи таймер NE555, можна побудувати мультівібратор, схема якого наведена на рис. 12.20.

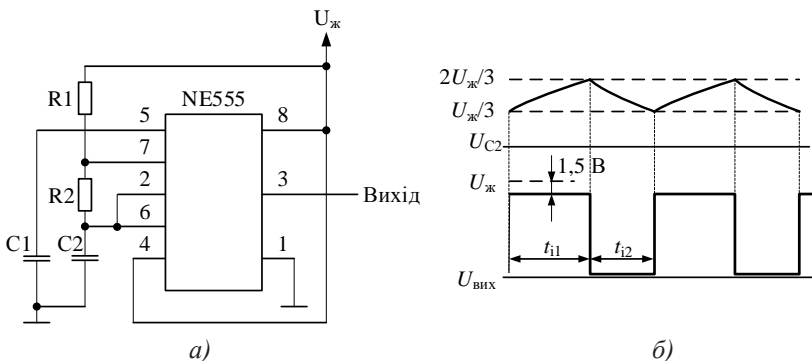


Рис. 12.20. Схема мультівібратора на базі таймера NE555 (а) та часові діаграми його роботи

Для покращення форми вихідних імпульсів резистор, що задає струм заряду конденсатора C2, розділений на дві частини. Після відкриття транзистора VT1 конденсатор C2 розряджається через резистор R2. Якщо транзистор VT1 закритий, конденсатор заряджається через послідовно з'єднані R1 і R2.

Якщо $U_C < U_{ж}/3$, включається нижній компаратор і виключається верхній (рис. 12.17), але коли $U_C > 2U_{ж}/3$, стан компараторів змінюється на протилежний. Відповідно сформованим компараторами сигналам перемикається тригер. Тому в моменти, коли $U_C > 2U_{ж}/3$, відкривається транзистор VT1 і конденсатор C2 починає розряджатися.

За час заряду конденсатора (на рис. 12.20 позначено t_{i1}) напруга U_C на ньому змінюється в межах $U_{ж}/3 < U_C < 2U_{ж}/3$. Оскільки U_C змінюється тільки у вказаних межах, то можна записати

$$\frac{U_{ж}}{3} + \frac{2U_{ж}}{3}(1 - e^{-t_{i1}/\tau_{зар}}),$$

де $\tau_{\text{зар}} = (R_1 + R_2)C_2$.

З попереднього рівняння неважко розрахувати тривалість імпульсу t_{i1} :

$$t_{i1} = \tau_{\text{зар}} \ln 2 = 0,693(R_1 + R_2)C_2.$$

Аналогічно можна визначити час розряду конденсатора через резистор R_2

$$t_{i2} = \tau_{\text{розр}} \ln 2 = 0,693R_2C_2.$$

Таким чином, період слідування імпульсів на виході генератора буде дорівнювати

$$T_{\text{вих}} = t_{i1} + t_{i2} = 0,693(R_1 + 2R_2)C_2,$$

а частота слідування імпульсів

$$f_{\text{вих}} = \frac{1}{t_{i1} + t_{i2}} = \frac{1,44}{(R_1 + 2R_2)C_2}.$$

Потрібно звернути увагу на те, що частота вихідного сигналу такого генератора від напруги живлення не залежить.

Шпаруватість імпульсів на виході (рис. 12.20) визначимо щодо тривалості позитивного вихідного імпульсу:

$$Q = \frac{t_{i1} + t_{i2}}{t_{i1}} = \frac{R_1 + 2R_2}{R_1 + R_2}.$$

Таким чином, у схемі рис. 12.20 неможливо отримати значення шпаруватості 2 (буде тільки при $R_1 = 0$, однак тоді схема стає непрацездатною).

13. ЗАПАМ'ЯТОВУВАЛЬНІ ПРИСТРОЇ

13.1. Оперативні запам'ятовувальні пристрої

Для короткочасного збереження невеликих об'ємів кодових слів зазвичай використовують регістри. При необхідності тривалого збереження або збереження великих об'ємів інформації застосовують запам'ятовувальні пристрої (ЗП), виконані на спеціалізованих ІС. Застосування ЗП, що використовують ІС, дозволяє максимально спростити апаратну частину електронних пристроїв.

За виконуваною функцією ЗП можна класифікувати на:

- оперативні запам'ятовувальні пристрої (ОЗП);
- постійні запам'ятовувальні пристрої (ПЗП).

До оперативних відносять ЗП, що використовуються для збереження інформації, отриманої в процесі роботи пристрою і забезпечують можливість зчитування та запису інформації протягом часу, коли до них підключене джерело живлення.

Оперативні ЗП можуть бути виконані як статичними, так і динамічними. У статичних ОЗП записана інформація постійно зберігається у виділеному для неї місці і не руйнується при її зчитуванні. Руйнування інформації можливо тільки при її примусовому стиранні або вимкненні напруги джерела живлення.

В динамічних ОЗП інформація постійно циркулює в масиві, відведеному для її збереження. При цьому зчитування інформації супроводжується її руйнуванням. Для збереження інформації її необхідно постійно відновлювати.

Для позначення на електричних схемах ІС ОЗП використовується скорочення RAM (Random Access Memory).

Варіант типової структурної схеми статичного ОЗП наведений на рис. 13.1. Для прикладу вибрана мікросхема пам'яті ємністю 256 біт. Структурна схема включає виконані на одному кристалі кремнію матрицю накопичувача (запам'ятовувальних елементів), дешифратори коду адреси рядків $A0...A3$ (DCx) і стовпців $A4...A7$ (DCy), пристрій управління (ПУпр), пристрій запису (ПЗап) і пристрій зчитування (ПЗчит). Режимом роботи пристрою керують сигнали \overline{CS} (Chip Select – вибір мікросхеми) і \overline{WR} / RD (Write/Read – запис-зчитування). Вхід, на який подається інформація, позначається як DI (Data Input), а вихід, з якого виводиться зчитувана інформація, DO (Data Output).

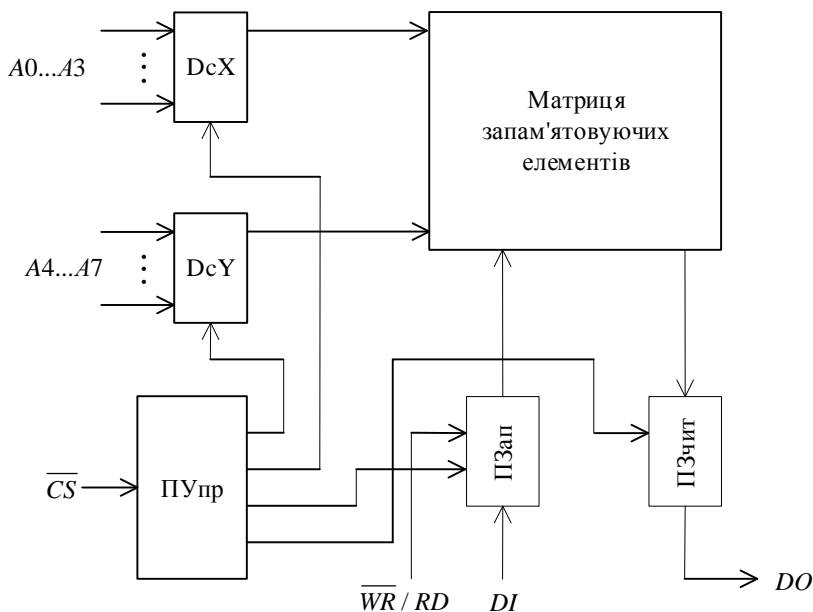


Рис. 13.1. Структурна схема запам'ятовувального пристрою з однорозрядною організацією

Матриця накопичувача містить 256 запам'ятовувальних елементів (ЗЕ), розташованих на перетинах 16 рядків і 16 стовпців (рис. 13.2). Кожен ЗЕ є статичним тригером, що може бути реалізований на основі біполярних або МДН-транзисторів. Це залежить від технології виготовлення мікросхем.

Доступ до кожного ЗЕ можна організувати за допомогою логічного елементу І (рис. 13.3).

Для збереження інформації в тригері необхідно джерело живлення, тобто тригер розглянутого типу є енергозалежним. При наявності живлення тригер здатний зберігати свій стан як завгодно довго.

Для запису 0 або 1 до мікросхеми необхідно підвести цю інформацію до входу DI , подати код адреси $A\{A_0 - A_7\}$, а також сигнал дозволу \overline{CS} і сигнал \overline{WR}/RD рівня лог. 0, що відповідає режиму запису. При зазначених сигналах збуджується заданий рядок X_i , вибраний дешифратором коду адреси рядків, в ре-

зультаті чого відкривається доступ за розрядними шинами до усіх ЗЕ даного рядка. Одночасно збуджується один з виходів Y_k дешифратора коду адреси стовпців і під'єднує вибраний ЗЕ.

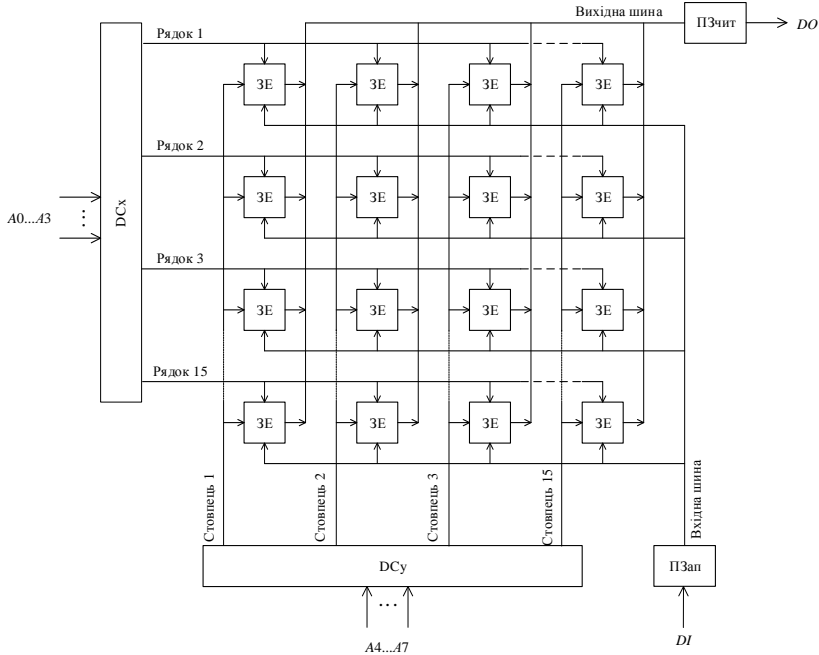


Рис. 13.2. Структура матриці запам'ятовувальних елементів

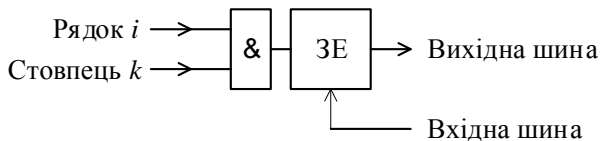


Рис. 13.3. Організація доступу до запам'ятовувальних елементів матриці

Звернення до мікросхеми для зчитування відбувається аналогічно, але при значенні сигналу \overline{WR} / RD , рівному лог. 1.

В більшості мікросхем пам'яті пристрій зчитування (або пристрій вводу/виводу) містить вихідний ключовий підсилювач-формувавч, здатний приймати три стани: два функціональних, відповідно лог. 0 або лог. 1, і третій стан, коли внутрішні блоки

запам'ятовувального пристрою відключаються від виходу. В третьому стані вихід відключений від приймача інформації, наприклад, інформаційної шини.

Наявність у мікросхеми виходу з трьома станами дозволяє з'єднувати інформаційні вхід і вихід для підключення їх до спільної (єдиної) інформаційної шини.

На рис. 13.4 наведені типові часові діаграми сигналів на входах статичного ОЗП.

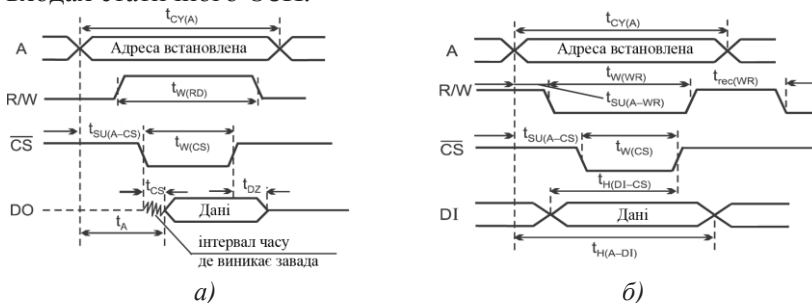


Рис. 13.4. Типові часові діаграми сигналів на входах статичного ОЗП в режимі читування (а) та запису інформації (б)

Розглянута структурна схема (рис. 13.1) є прикладом реалізації статичних ОЗП з однорозрядною організацією. Даний тип мікросхем переважає в сучасній номенклатурі мікросхем пам'яті. Разом з тим усе більш активний розвиток отримують мікросхеми статичних ОЗП з багаторозрядною організацією. Принцип побудови таких мікросхем аналогічний розглянутому вище, але при виборі адреси одного з ЗЕ відбувається звернення одразу до декількох ЗЕ, що мають одну адресу, але кожний під'єднаний до різних вхідних і вихідних шин. Пристрої запису і читування для цього мають n незалежних каналів, де n – кількість розрядів запам'ятовувального пристрою.

В мікросхемах пам'яті динамічного типу функції ЗП виконує електричний конденсатор, утворений всередині МДН-структури (рис. 13.5). Інформація в ньому представляється у виді заряду: наявність заряду на конденсаторі відповідає лог. 0, а відсутність – лог. 1. Оскільки час зберігання конденсатором заряду обмежений, передбачають періодичне відновлення (регенерацію) записаної інформації. В цьому полягає одна з відмін-

ностей динамічних ОЗП. Крім того, для них необхідна синхронізація, що забезпечує необхідну послідовність вмикань і вимикань функціональних вузлів.

Для виготовлення мікросхем динамічних ОЗП в основному використовують n -МДН-технологію, що дозволяє підвищувати швидкодію та рівень інтеграції мікросхем, забезпечувати малі струми витоку і за рахунок цього збільшувати час зберігання заряду на запам'ятовувальному конденсаторі.

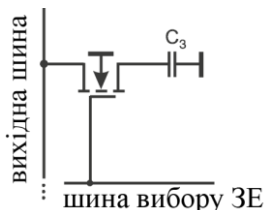


Рис. 13.5. Комірка динамічного ОЗП на МДН-структурах

Розглянемо типовий варіант реалізації динамічного ОЗП на прикладі мікросхеми інформаційною ємністю $16K \times 1$ біт. До складу її структурної схеми (рис. 13.5) входять виконані на одному кремнієвому кристалі матриця накопичувача, що містить 16384 елементів пам'яті, розташованих на перетинах 128 рядків і 128 стовпців, 128 підсилювачів зчитування і регенерації, дешифратори рядків і стовпців, пристрій управління, пристрій вводу-виводу та мультіплексний регістр адреси.

Матриця накопичувача розділена на дві частини по 64×64 ЗЕ в кожній. Між ними розміщені підсилювачі так, що кожний стовпець складається з двох секцій, підключених до різних плечей підсилювача.

Крім масиву ЗЕ та підсилювачів матриця має у своїй структурі опорні елементи (по одному елементу в кожній півшині). Ці елементи в кожній половині матриці складають опорний рядок. Опорний елемент побудований аналогічно запам'ятовувальному. Його призначення полягає в підтримуванні опорної напруги, з якою підсилювач порівнює потенціал півшини з вибраним ЗЕ та реагує на отриману різницю потенціалів при порівнянні (позитивного або негативного знаку залежно від рівня зчитування). Ця операція відбувається так: якщо вибрано для

звертання рядок верхньої півматриці X_i , то сигнал A_6 старшого розряду коду адреси рядка комутує в селекторі опорного рядка коло через ключовий транзистор для сигналу F_2 до опорного рядка, розташованого в нижній півматриці.

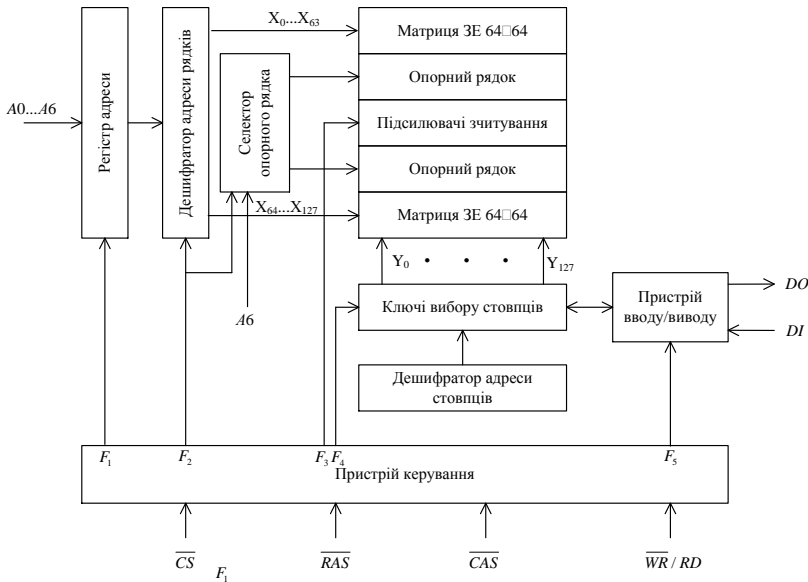


Рис. 13.5. Структура схема мікросхеми динамічного ОЗП

Таким чином, у кожному з 128 стовпців до підсилювача з різних боків підключені ЗЕ та опорні елементи. Оскільки потенціал півшини із ЗЕ відрізняється від опорного, то сигнал F_3 викликає перекидання тригера по переважному рівню. У підсумку на виходах-входах тригера формуються повні рівні лог. 1 та лог. 0. Той з сигналів, який відображає зчитану інформацію, комутується на вхід пристрою виводу. Очевидно, що зчитаним може бути тільки один сигнал з вибраного дешифратором стовпця.

Сигнал на вході-виході тригера-підсилювача виконує також функцію відновлення рівня заряду запам'ятовувального конденсатора, тобто функцію регенерації інформації. При цьому ця операція відбувається у всіх ЗЕ вибраного рядка одночасно.

Таким чином, при кожному звертанні до матриці для зчитування інформації автоматично здійснюється регенерація

інформації у всіх ЗЕ, що належать вибраному рядку.

Для адресації 16К елементів пам'яті необхідний 14-розрядний код. З метою зменшення кількості необхідних виводів корпусу в мікросхемах динамічних ОЗП код адреси вводять вроздріб: спочатку сім молодших розрядів $A_0 \dots A_6$, супроводжуючи їх сигналом стробу \overline{RAS} , а потім сім старших розрядів $A_7 \dots A_{13}$ з сигналом стробу \overline{CAS} . Всередині мікросхеми коди адреси рядків і стовпців фіксуються в адресному регістрі, дешифруються і здійснюють вибірку необхідного ЗЕ.

Для формування внутрішніх сигналів $F_1 \dots F_5$, керуючих вмиканням і вимиканням у визначеній послідовності функціональних вузлів мікросхеми, в її структурі передбачений пристрій управління, для якого входними є сигнали \overline{RAS} , \overline{CAS} та $\overline{WR} / \overline{RD}$.

Пристрій вводу-виводу забезпечує вивід одного біта інформації DO в режимі зчитування і ввід одного біта інформації DI з її фіксацією за допомогою D-тригера в режимі запису. В усіх режимах, крім режиму зчитування, вихід встановлюється у третій стан, що дозволяє поєднувати інформаційні вхід і вихід при підключенні мікросхеми до спільної інформаційної шини.

13.2. Мікросхеми масочних ПЗП

Мікросхеми ПЗП за способом програмування, тобто занесення в них інформації, діляться на три групи:

- ПЗП, що одноразово програмуються виробником за замовленням фотошаблоном (маскою), тобто масочні ПЗП (ПЗПМ, ROM);
- ПЗП, що одноразово програмуються користувачем при перепалюванні плавких перемичок на кристалі (ППЗП, PROM);
- ПЗП, що багаторазово програмуються користувачем, тобто репрограмовані ПЗП (РПЗП, EPROM).

Загальною властивістю всіх мікросхем ПЗП є їхня багаторозрядна організація, режим зчитування як основний режим роботи та енергонезалежність. Разом з тим у них є й істотні відмінності у способі програмування, режимах зчитування, а також

у зверненні до них при застосуванні. Тому доцільно розглянути кожену групу мікросхем ПЗП окремо.

Мікросхеми ПЗПМ виготовляють згідно біполярної ТТЛ, ТТЛШ-технології, а також n -канальної та p -канальної КМОН-технологій. Принцип побудови в більшості мікросхем групи ПЗПМ однаковий і може бути представлений структурою матриці запам'ятовувальних елементів (рис. 13.6). Структурна схема самої мікросхеми наведена вище (рис. 13.1). Матриця складається з масиву ЗЕ, кожний з яких розміщений на перетині рядка і стовпця. Елемент пам'яті ПЗПМ є резистивною або напівпровідниковою (діодною, транзисторною) перемичкою між рядком і стовпцем. На рис. 13.6 в якості перемичок використовуються діоди. Інформацію до матриці заносять в процесі виготовлення мікросхеми. У тих рядках, де на перетині рядка і стовпця встановлений діод, при зверненні до нього буде зчитуватися лог. 1, а там, де діодів немає, стовпець під'єднаний до спільного через резистор і з цього рядка зчитується лог. 0.

Мікросхеми на біполярних транзисторах програмують шляхом формування перемичок між рядками і стовпцями в тих точках матриці, куди варто занести лог. 1. У тих точках матриці, де має бути лог. 0, перемичку не формують.

Програмування мікросхем ПЗПМ здійснюють один раз. Оскільки схема з'єднань або граничні напруги транзисторів не залежать від режиму роботи мікросхеми, вона має властивість енергонезалежності. Завдяки цій властивості мікросхеми ПЗПМ широко використовують як носії постійних програм, підпрограм різного призначення, кодів фізичних констант, постійних коефіцієнтів та ін. Процес занесення в ПЗПМ інформації в технічній документації називають "прошиванням".

Мікросхеми ПЗПМ працюють в режимах зберігання і зчитування. Для зчитування інформації необхідно подати код адреси і сигнал управління (CS), що дозволяє цю операцію. Сигнали управління можна подавати рівня лог. 1, якщо вхід CS прямий, або лог. 0, якщо вхід інверсний. Частіше вхід CS виконується інверсним.

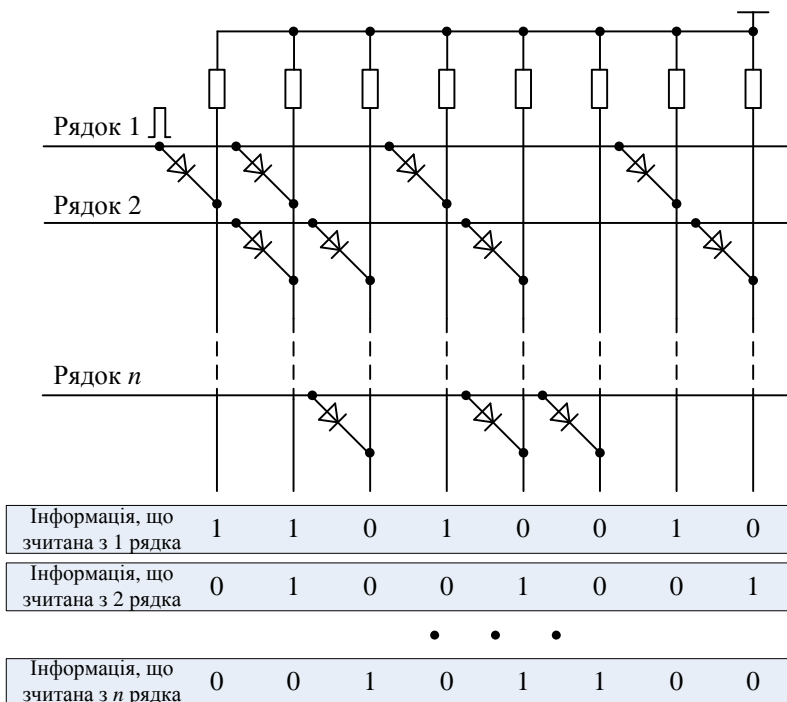


Рис. 13.6. Матриця запам'ятовувальних елементів мікросхеми масочного ПЗП

Багато мікросхем мають декілька входів управління, зазвичай зв'язаних визначеним логічним оператором. У таких мікросхемах на керуючі входи необхідно подавати визначену комбінацію сигналів для того, щоб сформувати умову дозволу зчитування.

Основним динамічним параметром мікросхем ПЗПМ є час вибірки адреси. За необхідності стробювання вихідних сигналів на входи керування CS варто подавати імпульси після надходження коду адреси. У такому випадку до розрахунку часу зчитування потрібно приймати час встановлення сигналу CS щодо адреси і час вибору.

Вихідні сигнали майже всіх мікросхем ПЗПМ мають TTL-рівні. Виходи побудовані в основному за схемою з трьома станами.

Для зменшення споживаної потужності деякі мікросхеми допускають застосування режиму імпульсного живлення. В цьому режимі живлення на мікросхему подають тільки при зчитуванні інформації.

13.3. Мікросхеми програмованих ПЗП

Мікросхеми програмованих ПЗП за принципом побудови і функціонуванням аналогічні масочним ПЗП, але мають відмінність у тому, що допускають програмування на місці свого застосування користувачем. Операція програмування полягає в руйнуванні частини плавких перемичок на поверхні кристалу імпульсами струму амплітудою (30...50) мА. Технічні засоби для виконання цієї операції досить прості і можуть бути виконані самим користувачем. Ця обставина в сполученні з низькою вартістю і доступністю мікросхем ППЗП обумовила їх широке практичне застосування.

Мікросхеми ППЗП, що випускаються українською промисловістю, здебільше виготовлені за ТТЛШ-технологією.

Типовий варіант структури та реалізації матриці запам'ятовувальних елементів мікросхеми ППЗП представлений на рис. 13.7. В основному вона повторює загальну структуру запам'ятовувальних пристроїв, але має додаткові пристрої $F_1 \dots F_n$ для формування струму програмування.

Матриця до програмування, тобто у початковому стані, містить однорідний масив провідних перемичок, що з'єднують рядки і стовпці у всіх точках їхніх перетинів. Перемички встановлюють з таких матеріалів: ніхрому, полікристалічного кремнію, силіциду платини та інших матеріалів. Перемичка в матриці виконує роль 3Е. Наявність перемички кодуєть лог. 1, якщо підсилювач зчитування (ПЗ) є повторювачем, і лог. 0, якщо підсилювач зчитування – інвертор. Отже, мікросхема ППЗП перед програмуванням залежно від характеристики вихідного підсилювача може мати заповнення матриці або лог. 0, або лог. 1 (частіше лог. 1).

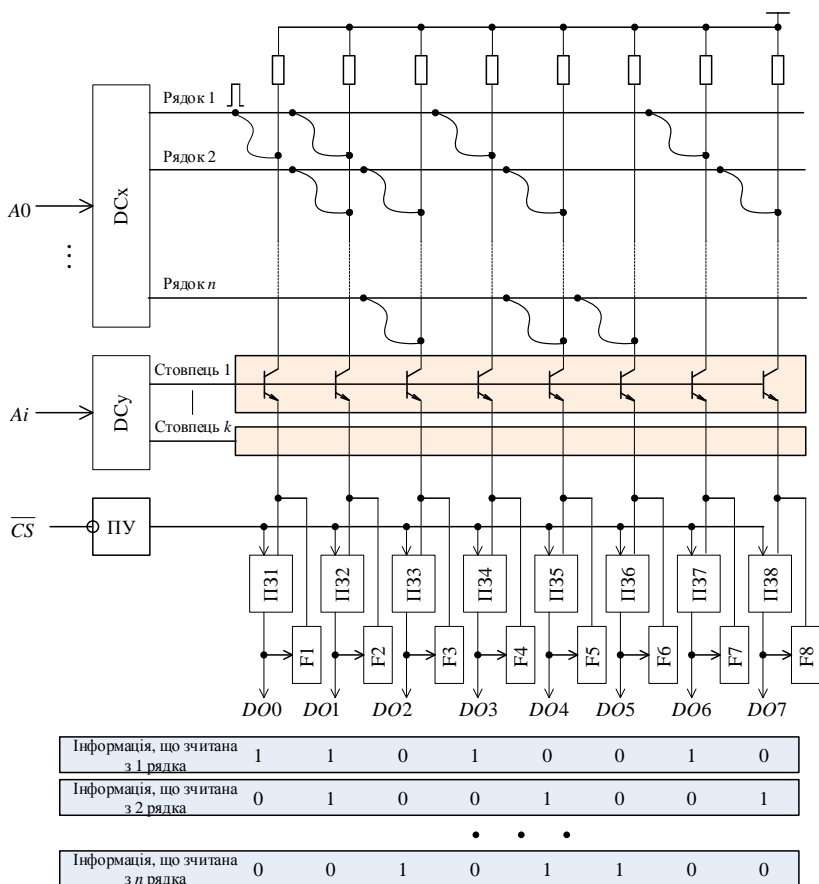
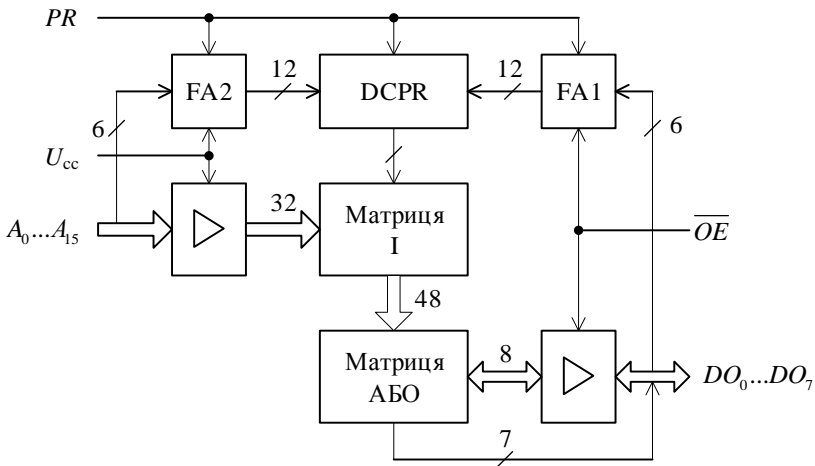


Рис. 13.7. Структура мікросхеми ППЗП

Робота запрограмованої мікросхеми ППЗП в режимі зчитування нічим не відрізняється від роботи мікросхеми ПЗПМ, розглянутих раніше. У деяких мікросхем є вивід для підключення джерела напруги програмування $U_{\text{п}}$. В режимі зчитування цей вивід не задіяний.

Різновидом ППЗП є програмовані випалюванням плавких перемичок логічні матриці (ПЛМ), виконані за ТТЛШ-технологією, що мають ідентичні характеристики і конструктивні параметри, але відрізняються типом виходу: у першій з мікрос-

Названі мікросхеми ПЛМ мають 16 входів $A_{15} - A_0$ для змінних, над якими ПЛМ виконує запрограмовані операції, вхід CS з нульовим рівнем дозволу, вхід PR дозволу запису, тобто програмування, і вісім виходів. Структура мікросхеми (рис. 13.8) включає операційну частину з матриці I, матриці АБО, вхідних і вихідних підсилювачів та програмуючу частину з адресними формувачами FA1, FA2 і дешифратори DCPR.



Оснoву ПЛМ (рис. 13.8) складають матриці І та АБО. Матриця І викoнує операції кон'юнкції над 16 вхідними змінними та їх інверсними значеннями, що надходять на рядкові шини матриці. Необхідні лoгічні добутки формують на шинax стoвпців шляхом випалювання непотрібних перемичок між рядками і стoвпцями.

Число стовпців 48, отже, на виході матриці I можна отримати до 48 логічних добутків, до кожного з яких може входити до 16 змінних та їх інверсій. Матриця АБО виконує операцію диз'юнкції над логічними добутками, сформованими матрицею I . Кількість виходів цієї матриці 8, тому вона здатна сформувати до восьми логічних сум, до кожної з яких може входити до 48 логічних добутків. Таким чином, можливості ПЛІМ харак-

теризуються кількістю точок комутації. Програмування матриці АБО виконується так само, як і матриці І, шляхом випалювання "непотрібних" перемичок. На виходах матриці АБО розміщені програмувальні підсилювачі, які залежно від стану перемички можуть передавати значення вихідної функції в прямій чи інверсній формі представлення.

Для програмування використовуються вбудовані вузли програмуючої частини, які збуджує сигнал дозволу PR. Програмування реалізують способом, аналогічним програмуванню ППЗП, за три етапи: спочатку програмують матрицю І, потім матрицю АБО та вихідні інвертори.

Функціональна схема ПЛМ наведена на рис. 13.9. Програмована логічна матриця має n входів, k елементів І, виходи яких утворюють k вертикальних шин, m елементів АБО, виходи яких підключені до входів суматорів за модулем 2 (М2), що виконують роль керованих інверторів. Виходи цих m інверторів є виходами самої ПЛМ. Кожен елемент І має $2n$ входів, якими він з'єднаний з усіма шинами вхідних сигналів та їх інверсій. В лінії зв'язку включені спеціальні перемички, позначені на рис. 13.9 короткими зигзагами. Ці перемички виконуються з певного матеріалу (наприклад, ніхром, кристалічний кремній) або у вигляді спеціальних p - n переходів так, щоб їх можна було вибірково руйнувати ("випалювати"), залишаючи лише ті зв'язки, які потрібні споживачу ПЛМ. У ряді типів ПЛМ випалювати перемички може сам споживач, подаючи на відповідні виводи корпусу імпульси струму або напруги певної амплітуди і тривалості.

Елементи АБО ПЛМ, так само як і елементи І, мають на входах випалювані перемички, за допомогою яких вони підключені до всіх вертикальних шин. Після випалювання на програматорі непотрібних перемичок у елементів АБО також залишаються лише ті зв'язки з вертикалями, які необхідні споживачу. Технічна реалізація елементів АБО така, що після випалювання перемичок на входах АБО, які ні до чого не підключені, забезпечуються рівні лог. 0.

Аналогічно програмують відсутність або виконання інвертування виходів АБО, відповідно перепалюючи або залишаючи перемички на верхніх на рис. 13.9 входах елементів М2.

Методи технологічного виконання елементів І, АБО, М2 і руйнування перемичок можуть бути різними. З точки зору логічного проектування суттєвим є лише те, що схемтехнік може використовувати ПЛМ на свій розсуд:

- подати на будь-який елемент І будь-яку комбінацію входів ПЛМ або їх інверсій;
- підключити до будь-якого елементу АБО будь-яку комбінацію вертикальних шин (виходів);
- проінвертувати вихідні сигнали АБО.

Такі можливості дозволяють дуже просто реалізувати на ПЛМ перетворювачі кодів або системи логічних функцій.

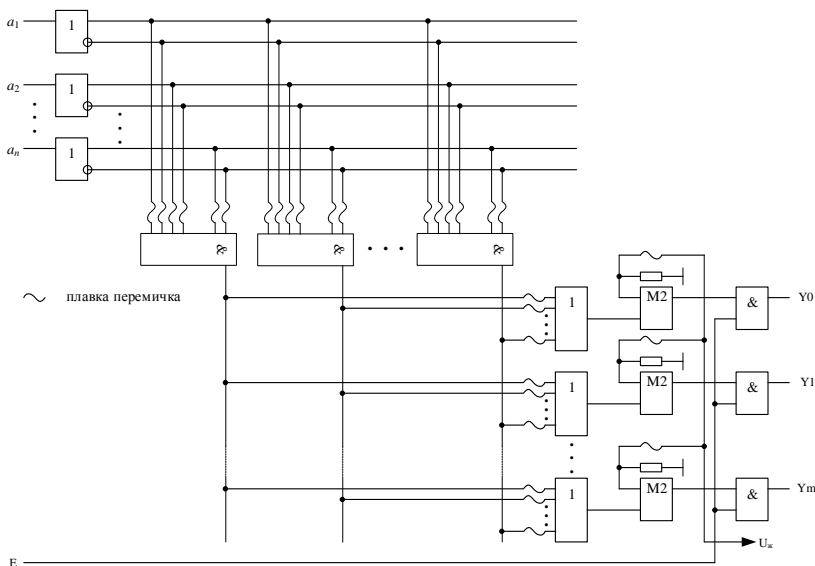


Рис. 13.9. Функціональна схема ПЛМ

Широко застосовуються ПЛМ, програмовані за способом замовленого фотошаблону на заводі-виробнику. Такі ПЛМ є різновидом масочних ПЗП. Вони включені, зокрема, до складу багатьох мікропроцесорних комплектів в якості ПЗП мікрокоманд. На основі ПЛМ можна будувати всілякі цифрові пристрої як комбінаційного, так і послідовнісного типів.

13.4. Мікросхеми репрограмованих ПЗП

Основна відмінна риса мікросхем репрограмованих ПЗП (РПЗП) полягає в їхній здатності до багаторазового (від 100 до 10 тис.) перепрограмувань самим користувачем. Ця властивість мікросхем забезпечена застосуванням ЗЕ з властивостями керованих "перемичок", функції яких виконують транзистори структури МНОН і транзистори n -МОН з плаваючим затвором (ПЗ) з використанням механізму лавинної інжекції заряду ЛІЗМОН.

Всю номенклатуру мікросхем РПЗП можна розділити на дві групи: РПЗП із записом і стиранням електричними сигналами (група ЕС) і РПЗП із записом електричними сигналами і стиранням ультрафіолетовим випромінюванням (група УФ).

Елементом пам'яті з структурою МНОН є МДН-транзистор з індукованим каналом p -типу (рис. 13.10, *а*) або n -типу, що має двошаровий діелектрик під затвором. Верхній шар формують з нітриду кремнію, нижній – з окислу кремнію, при цьому нижній шар значно тонший верхнього.

Якщо до затвору відносно підкладки прикласти імпульс напруги позитивної полярності з амплітудою (30...40) В, то при дії сильного електричного поля між затвором і підкладкою електрони набувають достатню енергію для того, щоб пройти тонкий діелектричний шар до межі розділу двох діелектриків. Верхній шар (нітриду кремнію) має значну товщину і електрони подолати його не можуть.

Накопичений на межі розділу двох діелектричних шарів заряд електронів знижує граничну напругу і зміщує передатну характеристику транзистора ліворуч (рис. 13.10, *б*). Цей стан ЗЕ відповідає лог. 1. Режим занесення заряду під затвор називають режимом програмування.

Рівню лог. 0 відповідає стан транзистора без заряду електронів у діелектрику. Щоб забезпечити цей стан, на затвор подають імпульс напруги негативної полярності з амплітудою (30...40) В. При цьому електрони витісняються в підкладку. При відсутності заряду електронів під затвором передатна характеристика зміщується в область високої граничної напруги. Режим витіснення заряду з підзатворного діелектрика називають режи-

мом стирання.

Режим стирання і програмування можна здійснити за допомогою напруги однієї полярності: негативної для p -МНОН або позитивної для n -МНОН структур. Ця можливість основана на використанні явища лавинної інжекції електронів під затвор, що відбувається, якщо до джерела і стоку прикласти імпульс негативної напруги (30...40) В, а затвор і підкладку з'єднати з корпусом. В результаті електричного пробою переходів джерело-підкладка і стік-підкладка відбувається лавинне розмноження електронів та інжекція деяких з них, які володіють достатньою кінетичною енергією, на границю між шарами діелектриків. Для стирання необхідно подати імпульс негативної напруги на затвор. В режимі зчитування на затвор подають напругу $U_{зч}$, значення якої лежить між двома граничними рівнями. Якщо в ЗЕ записана лог. 1, транзистор відкриється, а при лог. 0 – залишиться в закритому стані. В залежності від цього, як видно з рис. 13.11, z , в розрядній шині або буде протікати струм на вихід, або ні. Підсилювач зчитування трансформує стан шини в рівень напруги лог. 1 або лог. 0 на виході мікросхеми.

Для покращення характеристик РПЗП широко застосовують технологію виготовлення ЗЕ на n -МНОН транзисторах. Такі ЗЕ аналогічні розглянутому, але мають провідність підкладки p -типу, а джерело і стік n -типу.

Варіант ЗЕ на структурі ЛІЗМОН з подвійним затвором (рис. 13.11, e) є n -МОН транзистором, у якого в підзатворному однорідному діелектрику SiO_2 сформована ізольована провідна область з металу або полікристалічного кремнію. Цей затвор отримав назву "плаваючого".

В режимі програмування на керуючий затвор, джерело і стік подають імпульс напруги +(21...25) В. У зворотні зміщених p - n переходах виникає процес лавинного розмноження носіїв заряду і частина електронів інжектує на ПЗ. В результаті накопичення на ПЗ негативного заряду передатна характеристика транзистора зміщується в область високої граничної напруги (праворуч), що відповідає запису лог. 0.

Стирання записаної інформації, тобто витиснення заряду з ПЗ, в структурах ЛІЗМОН здійснюють двома способами: у

РПЗП-ЕС електричними сигналами, у РПЗП-УФ за допомогою УФ опромінення. В структурах зі стиранням електричними сигналами імпульсом позитивної напруги на керуючому затворі знімають заряд електронів з ПЗ, відновлюючи низьковольтний рівень граничної напруги, що відповідає лог. 1.

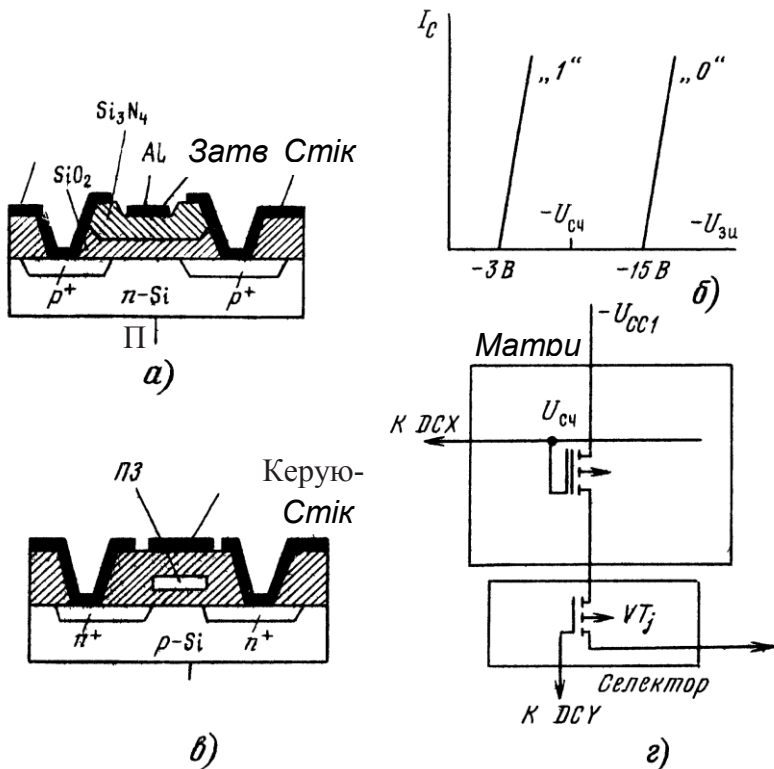


Рис. 13.11. Елементи пам'яті РПЗП типу МНОН (а) і його передатна характеристика (б), а також елементи РПЗП типу ЛІЗМОН (в) і розташування ЗЕ в матриці мікросхеми (г)

В структурах з УФ опроміненням електрони розсмоктуються з ПЗ у підкладку в результаті пришвидшення теплового руху за рахунок отриманої енергії від джерела УФ випромінювання. Режим зчитування здійснюють так само, як у ЕП на структурі МНОН. В режимі збереження забезпечують відсутність напруги на електродах ЗЕ для того, щоб виключити розсмокту-

вання заряду в діелектричному середовищі. Теоретичними розрахунками доведена можливість збереження заряду сотні років. На практиці цей час обмежують для одних типів мікросхем декількома тисячами годин, а для інших – декількома роками. Отже, мікросхеми РПЗП відносяться до групи енергонезалежних.

Принцип побудови і режим роботи РПЗП розглянемо на прикладі мікросхеми ємністю $2K \times 8$ із 3Е на *p*-МНОН транзисторах.

Структурна схема РПЗП (рис. 13.12) містить елементи, необхідні для роботи мікросхеми в якості ПЗП: матрицю з елементами пам'яті, дешифратори коду адреси рядків і стовпців, селектор (ключі вибору стовпців), а також пристрій вводу-виводу ПВВ. Крім того, в структурі передбачені функціональні вузли, що забезпечують її роботу в режимах стирання і програмування (запису інформації) – комутатори режимів і формувачі імпульсів необхідної амплітуди і тривалості з напруги програмування U_{PR} . У порівнянні з мікросхемами ПЗПМ і ППЗП система сигналів керування доповнена сигналами програмування PR і стирання ER. Накопичувач з матричною організацією містить 128 рядків і 128 стовпців, на перетинах яких розташовані 16384 елементів пам'яті. Управління накопичувачем здійснюють сімнадцять старшими розрядами адресного коду, який після дешифрування вибирає рядок з 128 елементами пам'яті. Сигнали, зчитані з елементів вибраного рядка, подаються на входи селектора, призначення якого складається у виборі з 128-розрядного коду на входах восьми розрядів, що далі надходять через ПВВ на виходи мікросхеми.

Селектором керують чотири молодші розряди адресного коду, які після дешифрування забезпечують вибірку одного восьмирозрядного слова з 16 слів, що містяться у вибраному рядку. Пристрій управління при дії сигналів на своїх входах забезпечує роботу мікросхеми в одному з таких режимів: зберігання, зчитування, стирання, запису (програмування). Сигнали керування мають таке призначення:

- CS – вибір мікросхеми;
- PR – дозвіл режиму запису (програмування);
- U_{PR} – напруга програмування;

- ER – сигнал зчитування;
- RD – сигнал стирання інформації.

Входи сигналів керування інверсні, тому значенням дозволу цих сигналів є рівень лог. 0. Більшість мікросхем групи ЕС допускають вибіркове стирання за вибраною адресою.

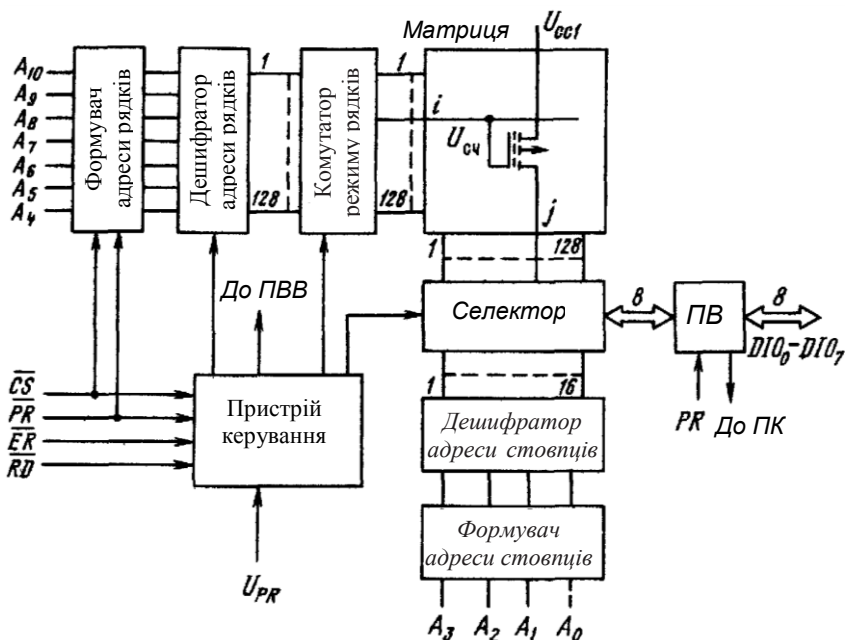


Рис. 13.12. Структурна схема мікросхеми РПЗП-ЕС

При експлуатації мікросхем РПЗП потрібно забезпечити необхідний порядок вмикання і вимикання напруги живлення окремих вузлів і програмування. При вмиканні спочатку подають напругу +5 В, потім –12 В і останньою напругу програмування. При вимиканні послідовність минається на зворотну. Можна всі три напруги включати і виключати одночасно.

Перевагою мікросхем РПЗП групи ЕС є можливість перепрограмування без вилучення їх з пристрою, де вони працюють. Іншою позитивною властивістю мікросхем даної групи є значна кількість циклів перепрограмування, що досягає для більшості мікросхем 10 тис. Ця їхня властивість у поєднанні з

енергонезалежністю дозволяє широко використовувати їх в апаратурі в якості вбудованих ПЗП зі змінюваною інформацією.

В структурі і режимах роботи мікросхем РПЗП-УФ багато спільного з мікросхемами групи ЕС. Виключення складає режим стирання, для реалізації якого необхідне джерело УФ випромінювання.

Подальший розвиток технології ПЛМ призвів до створення програмованих логічних інтегральних схем (ПЛІС). Реалізовані ними функції можна багаторазово змінювати, нерідко для цього навіть не потрібно витягати мікросхему з пристрою, в якій вона працює.

Кристал ПЛІС містить велику кількість однотипних логічних елементів. З'єднують елементи в потрібному порядку за допомогою перемичок – ключів на польових транзисторах. А в кожному логічному елементі є перемикачі (мультиплексори), програмуючи які можна змінювати виконувані функції.

Майже всі зовнішні виводи ПЛІС універсальні. Будь-який з них може бути або входом, або виходом. Деякі виводи мають додаткові функції, наприклад, використовуються для програмування конфігурації мікросхеми, що, однак, не виключає їхнього використання як звичайного входу або виходу.

За внутрішньою структурою і способом запам'ятовування конфігурації ПЛІС можна розділити на два види: *FPGA (field programmable gate array* – матриця вентилів, програмованих електричним полем) і *CPLD (complex programmable logic device* – складна програмована логічна матриця).

Мікросхеми *FPGA*, подібно ОЗП, зберігають задану конфігурацію тільки при включеному живленні і "забувають" її після вимикання. Щоразу, включивши живлення, таку ПЛІС необхідно програмувати заново. Програму зазвичай зберігають у встановленому на одній платі з ПЛІС завантажувальному ПЗП, що може бути послідовним однорозрядним або паралельним восьмирозрядним. Спосіб завантаження конфігурації задають логічними рівнями на входах керування. Процедура перезапису виконується автоматично, після чого ПЛІС переходить в робочий режим.

Мікросхеми *CPLD* зберігають конфігурацію незалежно

від наявності напруги живлення. До мікросхеми конфігурацію заносять за допомогою програматора, а стирають ультрафіолетовим випромінюванням. Такі ПЛІС встановлюють, як правило, у вже настроєні пристрої. Для настроювання нових виробів вони незручні через тривалу (до години) процедуру очистки.

Мікросхеми, виконані за технологією FLASH, можна неодноразово програмувати, не вилучаючи з пристрою, в якому вони встановлені. Для цього в них передбачений спеціальний порт JTAG. Щоб записати вихідну конфігурацію або внести до неї зміни, виводи цього порту через кілька буферних мікросхем з'єднують кабелем з портом LPT персонального комп'ютера.

Порт JTAG корисний не тільки для програмування. Через нього комп'ютер може отримати інформацію про логічні рівні на усіх виводах ПЛІС і в контрольних точках всередині її. Можливість організувати такий порт передбачений і в ПЛІС структури FPGA останніх серій. Виводи мікросхеми, зайняті портом JTAG, можуть бути і звичайними входами/виходами.

Контрольні питання

1. Для чого призначені ПЗП?
2. Як організовані схеми ПЗП і на яких ЗЕ вони реалізуються?
3. Як можна задавати лог. 0 і лог. 1 у матричних ПЗП?
4. Як програмуються однократно програмовані ПЗП?
5. На яких елементах виконуються РПЗП?
6. Поясніть принцип роботи ЗЕ РПЗП.
7. Як здійснюється запис інформації в РПЗП?
8. Перерахуйте цифрові схеми, які можна віднести до програмованих логічних пристроїв.
9. Назвіть основні складові частини ПЛІС.
10. Чим пояснюється функціональна гнучкість ПЛІС?
11. Наведіть основні параметри ЗП.
12. Що таке елементарна запам'ятовувальна комірка?
13. У чому полягає принцип побудови ІС ОЗП з різними типами адресації?
14. Наведіть структуру ІС динамічного ОЗП.
15. У чому полягає особливість динамічних ОЗП?

14. ЗАСТОСУВАННЯ ЦИФРОВИХ ІНТЕГРАЛЬНИХ МІКРОСХЕМ

14.1. Завади та завадостійкість цифрових пристроїв

В процесі вдосконалення пристроїв цифрової техніки вони мають все більше застосування для автоматизації управління різними технологічними об'єктами. При цьому в безпосередній близькості один від одного знаходяться електронні пристрої та потужні агрегати, такі як електродвигуни, електромагніти, електронагрівальні прилади, зварювальні машини та ін. В зв'язку з цим усе більшого значення набуває проблема зменшення шумів і забезпечення завадостійкості електронних пристроїв. Ця проблема пов'язана не тільки зі збільшенням рівня зовнішніх завад від розташованих поблизу агрегатів, але й зі збільшенням внутрішніх завад – взаємного впливу елементів і блоків електронних пристроїв один на одного. Останнє є особливо суттєвим в зв'язку з тенденцією підвищення швидкодії та мікромініатюризацією електронних схем.

Цифровий пристрій є мережею з'єднаних між собою елементів. Електричне коло, що з'єднує вихід одного елементу (джерела сигналу) з входом іншого елементу (приймача сигналу), утворює контур, що є свого роду антеною, яка перетворить зміну електромагнітного поля в ЕРС і струм у контурі. Під дією цієї ЕРС на вході елемента-приймача виникає завада, що залежить від площі контуру, швидкості зміни електромагнітного поля, вихідного опору елемента-джерела і вхідного опору елемента-приймача. Завади, створювані електромагнітним полем, розглядають для досить віддалених джерел, якими зазвичай є порівняно потужні агрегати. Крім них є також джерела завад, розташовані в безпосередній близькості від вхідних кіл елементу. Це сусідні провідники, що мають з розглянутим колом розподілений індуктивний та ємнісний зв'язок. Завади, обумовлені впливами через ці зв'язки при передачі сигналів по сусідніх колах, називають перехресними.

Елементи, що входять до складу цифрового пристрою, підключаються до спільних джерел живлення. При цьому створюються завади через спільний вихідний опір $R_{ж}$ джерела живлення й опір шин $R_{ш}$, що підводять живлення (рис. 14.1).

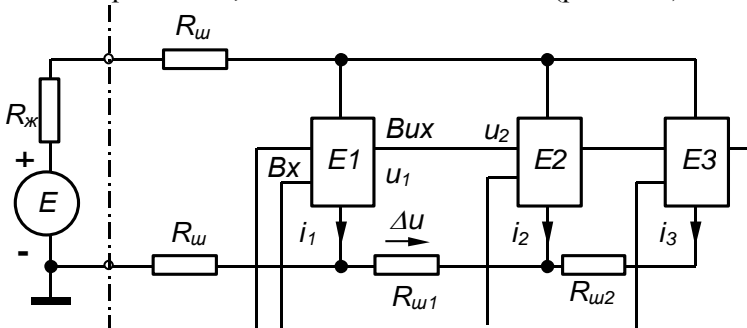


Рис. 14.1. Зв'язок елементів через спільне джерело живлення та шини підключення

Шини, що підводять живлення, мають розподілені параметри: ємність, індуктивність та опір. Для прикладу на рис. 14.2 наведена еквівалентна схема шини живлення елементів транзисторно-транзисторної логіки (ТТЛ).

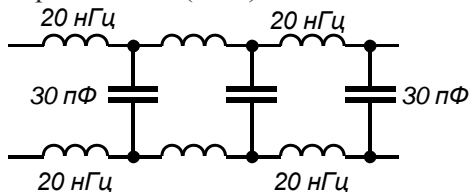


Рис. 14.2. Еквівалентна схема шини живлення логічних елементів

При цьому прийнято, що індуктивність провідника, який з'єднує виводи живлення двох поруч розташованих мікросхем, становить 20 нГн, а ємність мікросхеми між виводом живлення і виводом "землі" дорівнює 30 пФ. Хвильовий опір такої системи шин з підключеними мікросхемами становить 36 Ом. На рис. 14.3 показана типова діаграма імпульсів завад в такій шині живлення. Спад напруги на спільних для всіх елементів опорах $R_{ж}$ і $R_{ш}$ (рис. 14.1) змінює напругу, яка підводиться до елементів. Ці збільшення напруги живлення з деяким послабленням передаються на вихід елемента і створюють заваду. Найбільші завади

створюються опорами ділянок нульової (спільної) шини. Так, з рис. 14.1 видно, що спад напруги Δu на опорі R_{III} є завадою, що діє в колі передачі сигналу з виходу елементу E1 на вхід елементу E2.

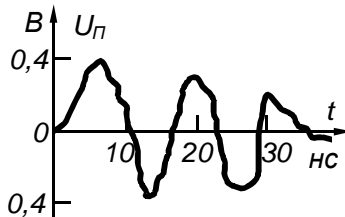


Рис. 14.3. Діаграма імпульсів завад в шинах

Завадостійкість логічного елементу є мірою його здатності не реагувати на помилкові вхідні сигнали, викликані електромагнітними наводками у вхідному колі, перехресними завадами і завадами в шинах живлення. Оскільки всі типи логічних елементів складаються з інерційних компонентів (транзистори, діоди, магнітні осердя), для перемикавання яких потрібна певна енергія вхідного сигналу, а запас завадостійкості елементу для тривалих і короточасних завад неоднаковий.

Підвищення завадостійкості пристроїв можна забезпечити, з одного боку, вибираючи елементи підвищеної завадостійкості, а з іншого боку – зменшуючи рівень завад на входах елементів за рахунок відповідних конструктивних і схемних заходів.

Завадостійкість елементу тим вище, чим більші вхідні напруги і струм необхідні для його перемикавання і чим більший час перемикавання. Узагальнюючи ці показники, можна розглядати енергію, необхідну для перемикавання. Зокрема, енергія перемикавання магнітних елементів з діодами на 2...3 порядки вище енергії перемикавання потенційних елементів інтегральних мікросхем. Однак при виборі елементної бази доводиться брати до уваги й інші фактори: швидкодію, вартість, рівень інтеграції, номенклатуру операційних елементів і функціональних блоків, а також технологічність виробництва апаратури. Для зменшення завад використовують такі заходи: зменшення площі контуру кіл, екранування з'єднань, заземлення, гальванічну розв'язку кіл,

фільтрацію, обмеження швидкості перемикачів струму в колах, розв'язку по живленню, екранування блоків і пристроїв.

Для зменшення завад, що виникають в цифрових логічних схемах, необхідно працювати з сигналами, що мають великий час наростання і спаду, малу амплітуду, обмежити кількість сигналів, що одночасно перемикаються, та застосовувати ефективні методи шунтування і заземлення. Для підвищення стійкості схем до зовнішніх завад варто застосовувати повільні схеми синхронізації з тригерами Шмітта на вході. Якщо система має довгі кабелі, бажано використовувати диференціальні передавальні і приймальні пристрої, з'єднані симетричними лініями зв'язку для зменшення рівня створюваних і прийнятих завад.

Шунтувальні конденсатори – джерела імпульсного струму, споживаного цифровими схемами при перемиканні, зменшують спад напруги в колах живлення та заземлення і сприяють фільтрації завад, які створюють джерела живлення і заземлення. На рис. 14.4, *а* показаний традиційний простий спосіб шунтування, що застосовується в цифрових схемах. Оксидний шунтувальний конденсатор великої ємності ((10...100) мкФ, 1 мкФ на кожен схему) розташований поблизу джерела живлення. Поруч з кожною тригерною ІС знаходиться керамічний шунтувальний конденсатор ємністю 0,1 мкФ, а до швидкодіючих схем приєднуються також керамічні шунтувальні конденсатори малої ємності (100...1000) пФ. Кожна схема, що передає сигнали за межі друкованої плати, на якій вона знаходиться, або що приймає сигнали від інших друкованих плат, має керамічний шунтувальний конденсатор ємністю 0,1 мкФ між джерелом живлення і лінією зворотного струму сигналу (рис. 14.5). До інших ІС підключаються керамічні шунтувальні конденсатори ємністю (0,01...0,1) мкФ, при цьому на 5 ІС приходить щонайменше один конденсатор. Недолік такого способу шунтування полягає в тому, що шунтувальні конденсатори слабо фільтрують завади з частотою вище 10 МГц через низькі власні резонансні частоти.

Рис. 14.4, *б* ілюструє іншу схему шунтування. Конденсатори С1, С2 та феритове кільце утворюють П-подібний фільтр, що послаблює високочастотні завади в друкованій платі. До кожної схеми, що передає сигнали за межі друкованої плати або

приймає сигнали, що надходять на плату ззовні, приєднується керамічний шунтувальний конденсатор ємністю 0,1 мкФ, який розташовується між джерелом живлення $U_{дж}$ і лінією зворотного струму (рис. 14.5). Всі інші IC забезпечуються керамічними шунтувальними конденсаторами ємністю

$$C \geq 4,5nC_H (\Phi),$$

де n – кількість виходів IC, C_H – ємність навантаження.

Поблизу джерела живлення міститься танталовий оксидний конденсатор для фільтрації низькочастотних завад. Ємність останнього має принаймні в 10 разів перевищувати ємність інших конденсаторів у колі, разом узятих.

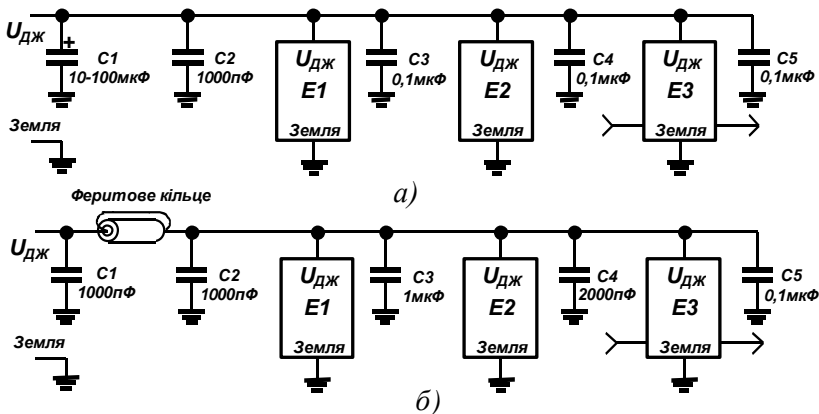


Рис. 14.4. Способи шунтування шин живлення цифрових мікросхем

Таке розташування компонентів забезпечує задовільне шунтування до частот 100 МГц і більше. Керамічні конденсатори повинні мати еквівалентну послідовну індуктивність менше 20 нГн і еквівалентний послідовний опір менше 0,5 Ом. З урахуванням індуктивності виводів танталові (або полікарбонатові) конденсатори повинні мати еквівалентні послідовні індуктивність і опір відповідно менше 30 нГн і 1 Ом. Імпеданс феритового кільця при тактовій частоті схеми не має перевищувати 10 Ом, а на частотах, що перевищують тактову частоту IC у 5 разів, імпеданс має перевищувати 50 Ом при робочому струмі. Найкращі результати виходять, коли провідник двічі пропускається крізь феритове кільце. Якщо імпеданс одного феритового кільця

занадто малий, можна застосовувати послідовно кілька кілець або більш товсте кільце. Якщо схема допускає додатковий спад напруги, замість феритового кільця можна використовувати композиційний резистор опором близько 51 Ом.

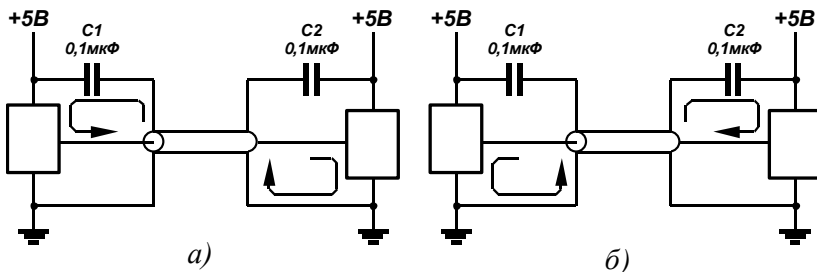


Рис. 14.5. Шунтування елементів, які передають та приймають сигнали за межі друкованої плати при переході від нижнього до верхнього логічного рівня (а) та при переході від верхнього до нижнього логічного рівня (б)

В діапазоні частот 30 МГц...1 ГГц синхронізаційні тактові сигнали та їхні гармоніки є основною причиною випромінюваних завад. Парні гармоніки можна різко зменшити, якщо використовувати сигнали синхронізації з 50% коефіцієнтом заповнення (коли тривалість сигналів синхронізації приблизно дорівнює паузі між ними). Також варто спробувати зменшити кількість ІС, керованих кожним тактовим сигналом. Якщо тактові сигнали мають надходити на кілька плат, як буфери бажано використовувати вхідні логічні елементи на тригерах Шмітта, а також обмежити розмах напруги і швидкість наростання (dU/dt) амплітуди основних сигналів синхронізації. Якщо сигнали синхронізації керуються перемикачами поза платою, потрібно уникати безпосереднього управління сигналами синхронізації від цих перемикачів. Замість цього бажано приєднати до перемикачів кола, які керують логічними елементами на платі, що у свою чергу будуть керувати тактовими сигналами. Зменшенню завад від тактових сигналів будуть сприяти також їх рознесення за фазою та децентралізація.

Проблему завад можна розв'язати шляхом ретельної синхронізації системи. Для зменшення перехідних струмів, що ви-

никають в джерелах живлення і пристроях заземлення, варто керувати невеликою групою мікросхем за допомогою рознесених тактових сигналів. Скороченню інтервалу часу, протягом якого система найбільш піддана впливу завад, сприяє застосування схем синхронізації вхідних сигналів (крім того бажано проводити стробування даних).

Кожний вхідний сигнал, що надходить на плату, варто подавати тільки на одну ІС (бажано, з тригерами Шмітта на входах), що допоможе розв'язати проблеми синхронізації. Вхідні кола на тригерах Шмітта підвищують заводозахищеність схем і полегшують обробку сигналів з повільним наростанням амплітуди. Якщо вхідний сигнал подається на стандартний логічний елемент, час наростання і спаду варто підтримувати малими для запобігання утворення паразитних коливань.

Особливої увагу вимагають сигнали, що виходять з плати. Виходи тригерів, лічильників та регістрів зсуву необхідно доповнювати буферними логічними елементами або підсилювачами-передавачами для розв'язання проблеми відбиття та викидів від неузгоджених ліній зв'язку. Якщо провідники, що виходять з плати, пропустити через феритове кільце, то це дає можливість підключати до виходу неузгоджені лінії зв'язку більшої довжини. Сигнали, що виходять з плати, не мають надходити на вхід схем, що знаходяться на цій платі. Якщо зневажити цим правилом, то можна зіткнутися з дуже серйозними проблемами наявності завад, що виникають на вихідному передавачі.

Для запобігання відбиття та викидів довгі лінії зв'язку мають узгоджуватися відповідно до їхнього характеристичного імпедансу. В цьому випадку корисним може виявитися послідовно підключений резистор за умови, що всі приймальні пристрої знаходяться на дальньому кінці лінії зв'язку (рис. 14.6, *а*). Резистор варто підібрати таким чином, щоб при перемиканні сигналу в лінії зв'язку виникали лише незначні викиди. Подільник напруги на резисторах на дальньому кінці лінії зв'язку дозволить розмістити приймачі в будь-якій її місці (рис. 14.6, *б*). Для зменшення завад в довгій кабельній мережі варто застосовувати передавальні та приймальні пристрої, що формують диференціальні сигнали, які передаються по симетричних лініях зв'язку.

В односпрямованих лініях зв'язку необхідно підключати навантаження до дальніх кінців (рис. 14.7, а), а в двоспрямованих – до обох (рис. 14.7, б). В симетричних лініях зв'язку з належним чином узгодженим навантаженням придушення завад може досягти 70 дБ (3000:1) в діапазоні частот (0...100) кГц.

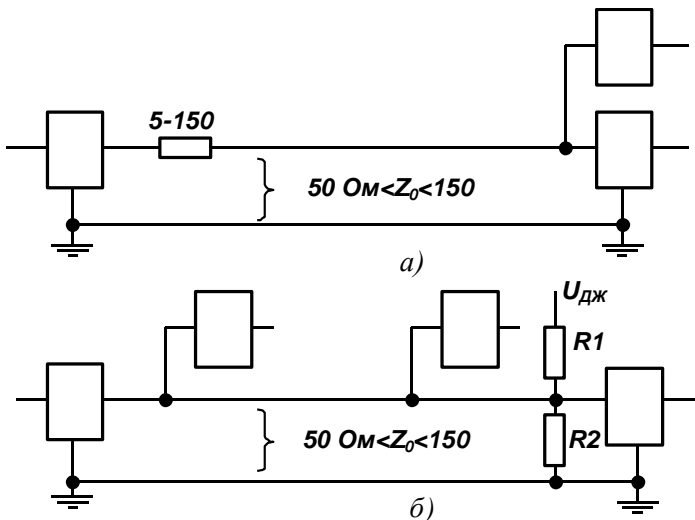


Рис. 14.6. Схеми узгодження довгих ліній зв'язку для зменшення відбиття сигналів

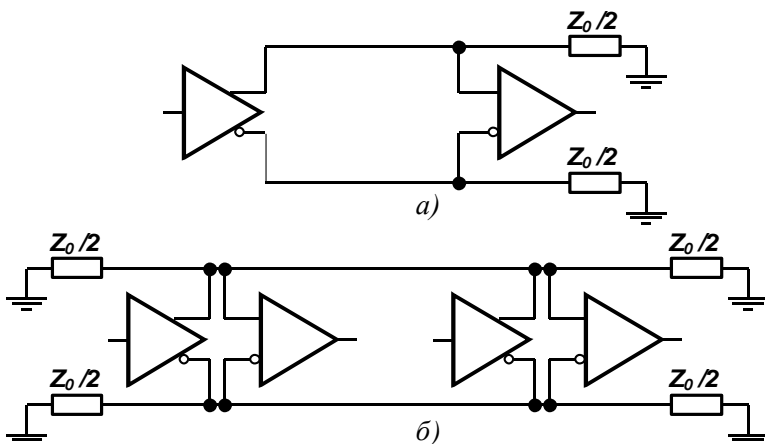


Рис. 14.7. Схеми підключення навантаження до симетричних ліній зв'язку: односпрямованих (а) та двоспрямованих (б)

Підключення узгодженого навантаження до кабелів, контактних площадок друкованих плат та проводів зв'язку має проводитися завжди, якщо час поширення сигналу перевищує половину часу наростання або спаду його амплітуди. Імпеданс узгодження плоских і коаксіальних кабелів, а також скручених пар становить 100 Ом ($75 \text{ Ом} \leq Z_0 \leq 120 \text{ Ом}$). Імпеданс узгодження контактних площадок друкованих плат має дорівнювати їхньому характеристичному імпедансу ($20 \text{ Ом} \leq Z_0 \leq 200 \text{ Ом}$).

Подільник на резисторах (рис. 14.8, а) може бути узгодженим навантаженням і подавати напругу зсуву на лінію зв'язку без додаткових джерел живлення. Наприклад стандартні узгоджені навантаження для ТТЛ-схем дорівнюють $R_1 = 330 \text{ Ом}$ і $R_2 = 220 \text{ Ом}$, при цьому з вихідного резистора R_2 знімається напруга +2 В ($U_{дж} = +5 \text{ В}$). При $R_1 = 150 \text{ Ом}$ і $R_2 = 470 \text{ Ом}$ з вихідного резистора знімають напругу +3,8 В, а при $R_1 = 120 \text{ Ом}$ і $R_2 = 470 \text{ Ом}$ – напругу +4 В.

Більшість КМОН ІС мають навантажувальну здатність, недостатню для підключення подільників на резисторах. До них можна підключати резистивно-ємнісний подільник, показаний на рис. 14.8, б, де $R_3 = 1 \text{ кОм}$, $R_4 = 330 \text{ Ом}$, $C_1 = 1000 \text{ пФ}$.

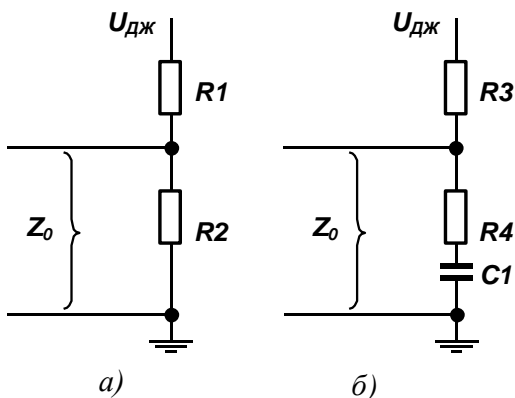


Рис. 14.8. Узгодженні навантаження для ліній зв'язку схем ТТЛ (а) та КМОН (б)

Всі невикористані входи ІС мають перебувати або під

високим, або під низьким потенціалом. Деякі системи передбачають можливість відключення невикористаних вузлів.

14.2. Рекомендації з монтажу інтегральних мікросхем

Однією з основних умов, що забезпечують надійну роботу електронної апаратури, сконструйованої із застосуванням цифрових інтегральних мікросхем, є дотримання правил щодо їх установки, пайки та монтажу.

Основними елементами конструкцій вузлів і блоків електронної апаратури, що дозволяють найбільш повно реалізувати переваги ІС, є друковані плати. Друковані плати використовуються в масовому виробництві, а також для створення швидкодіючих та малогабаритних систем. З їхньою допомогою легко проводити складання систем, вони компактні і мають високі частотні характеристики. Їхні основні недоліки – тривалий час конструювання і труднощі, з якими доводиться зіштовхуватися при внесенні конструктивних змін.

Конструювання монтажних плат можна умовно розбити на вісім основних етапів:

- визначення розмірів і форми плати, а також положення з'єднувачів;
- визначення розташування ІС;
- розміщення на платі кіл землі та живлення;
- розміщення дискретних компонентів;
- визначення топології кіл землі та живлення;
- розкладка кіл синхронізації;
- розкладка інших сигнальних кіл;
- остаточне виконання проекту.

Розмір і форму друкованих плат, а також положення з'єднувачів можна визначити, виходячи з фізичного проекту виробу.

При розробці конструкцій друкованих плат приймається ряд принципових рішень, пов'язаних з вибором таких параметрів, як кількість позицій m для установки ІС, крок установки ІС в горизонтальному і вертикальному напрямках h_x , h_y , кількість зовнішніх виводів q , а також кількість прошарків для реалізації

з'єднань s . Вибір цих параметрів пов'язаний з врахуванням економічних, технологічних, схематичних, технічних чинників і здійснюється на основі дослідних даних. При цьому порушення оптимальних співвідношень між параметрами істотно ускладнює процес конструювання і збільшує вартість розробки апаратури.

Необхідну кількість зовнішніх виводів можна оцінити за формулою $q = cn$, де c – середня кількість виводів ІС; n – кількість ІС на платі; $r = 0,57 \dots 0,75$. Найбільше значення r відповідає операційним пристроям паралельного типу, найменше – комбінаційним логічним схемам.

Крок установки ІС в горизонтальному і вертикальному напрямках, що виражається кількістю кроків між друкованими провідниками, можна знайти за формулами

$$h_x = b + \frac{n_y c}{4};$$

$$h_y = a + \frac{n_x c}{4},$$

де n_x – кількість ІС в ряду; n_y – кількість ІС у стовпчику; c – середня кількість виводів у ІС; a, b – розміри модуля уздовж вертикальної та горизонтальної осей плати, що виражаються числом кроків між друкованими провідниками.

Питома площа плати (на один модуль):

$$s_1 = h_x h_y = ab + c(bn_x + an_y)/4 + (cn_x n_y)/16.$$

При збільшенні кількості модулів $n_x n_y$ питома площа s_1 збільшується. При заданій кількості елементів площа плати буде мінімальна, якщо $n_x = an_y/b$. За цих умов конфігурація плати наближається до квадрату. Крім того, така форма сприяє зменшенню довжини провідників та полегшує її конструювання і виготовлення.

Максимальний розмір сторони друкованої плати, як односторонньої, так і багаторонньої, не має перевищувати 500 мм. Це обмеження обумовлюється вимогами міцності та щільністю монтажу: чим більше плата, тим менше щільність монтажу. На практиці зазвичай друковані плати з стороною до 100 товщин матеріалу діелектрика виготовляють без додаткових деталей, що

підвищує її жорсткість. Для плат з більшими розмірами передбачають спеціальні заходи підвищення жорсткості (додаткові точки кріплення у пристрої, введення ребер жорсткості та ін.). Співвідношення розмірів сторін друкованої плати для спрощення компоновання блоків та уніфікації розмірів друкованих плат рекомендуються приймати 1:1, 2:1, 3:1, 4:1, 3:2, 5:2 та ін. Наведені оцінки можуть бути вихідною базою для вибору основних параметрів плати.

На другому етапі необхідно вирішити, де розмістити ІС. Від правильного розташування корпусів мікросхем на друкованих платах залежать такі параметри приладу, як габарити, маса та надійність роботи. Чим щільніше будуть розташовуватися корпуси мікросхем на площині друкованих плат, тим складніше автоматизувати їх монтаж, тим більш жорстким буде температурний режим їхньої роботи, тим більший рівень завад буде наводитися в сигнальних провідниках. І навпаки, чим більше відстань між мікросхемами, тим менш ефективно використовується фізичний об'єм, тим більшою є довжина ліній зв'язків. Тому при установці мікросхем на друкованій платі варто враховувати всі наслідки вибору того або іншого варіанту їхнього розміщення.

Вибір кроку установки мікросхем на друкованій платі обумовлюється необхідною щільністю компоновання мікросхем в апаратурі, температурним режимом роботи, методом розробки топології друкованих плат (ручний, машинний), складністю електричної принципової схеми та конструктивних параметрів корпусу мікросхеми. Незалежно від типу корпусу крок установки мікросхем рекомендується приймати кратним 2,5 мм. При цьому зазори між корпусами не мають бути менше 1,5 мм. У технічно обґрунтованих випадках крок установки мікросхем може бути прийнятий кратним 1,25 мм.

Необхідно дотримуватись лінійно-багаторядного розташування ІС на друкованих платах. Однак допускається їхнє розміщення в шаховому порядку. Таке розташування корпусів мікросхем дозволяє автоматизувати процеси зборки і контролю, з більшою ефективністю використовувати корисну площу друкованої плати і прямокутну систему координат для визначення місця розташування корпусів.

Мікросхеми з штирьовими виводами в корпусах мають встановлюватися тільки з одного боку друкованої плати. Це пояснюється тим, що монтаж штирьових виводів, як правило, роблять у наскрізні металізовані отвори, при цьому кінці виводів виступають на зворотному боці плати. Установку мікросхем в корпусах з штирьовими виводами на друковану плату роблять із зазором (1,0...2,0) мм або з електроізоляційною прокладкою товщиною (1,0...1,5) мм, попередньо приклеюючи їх до поверхні друкованої плати. Прокладку варто розміщувати під усією площею основи корпусу або між виводами на площі не менше 2/3 основи, при цьому вона має виключати можливість торкання виступаючих виводів. Зазор між корпусом ІС і платою має бути не більше 1,5 мм; зазор між корпусами ІС має бути не менше 1,6 мм; виступаючі частини виводів мають знаходитись над поверхнею плати в межах 0,5...1 мм (якщо в ТУ не обговорено інше). Рекомендується крок установки мікросхеми за вузьким боком корпусу 10 мм, а за широким боком – 25 мм (рис. 14.9).

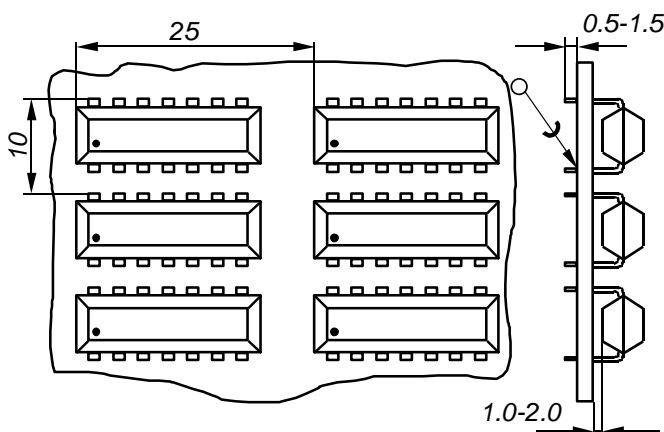


Рис. 14.9. Схема установки мікросхем з штирьовими виводами на друковану плату

При установці ІС на друковані плати часто виникає необхідність формовки виводів. Вимоги, що ставляться до формовки, обумовлюються в технічній документації. Для мікросхем з штирьовими виводами формування, як правило, має проводитися з радіусом вигину не менше $2d$ (d – діаметр виводу) і відстан-

ню від корпусу мікросхеми до центру кола вигину не менше 1 мм (якщо в ТУ не вказано інше). Штирьові виводи утримують корпус мікросхеми досить міцно і витримують практично будь-які механічні впливи.

Мікросхеми з планарними виводами також рекомендуються встановлювати з одного боку друкованої плати. Лише в технічно обґрунтованих випадках допускається їх встановлювати з обох боків плати. Існує три варіанти установки корпусів мікросхем в корпусах з площинними виводами на друкованій платі (рис. 14.10, а-в): без зазору (формування за рис. 14.10, з) та з зазором і на прокладку (формування за рис. 14.10, д).

При монтажі без зазору корпус мікросхеми встановлюють на друковану плату, виводи орієнтують щодо контактних площадок і припаюють. Корпус утримується на виводах або ж його приклеюють до поверхні плати нітроклеєм або епоксидним клеєм. В деяких випадках допускається установка ІС впритул на платі або із зазором не більше 0,7 мм (якщо в ТУ не вказано інше).

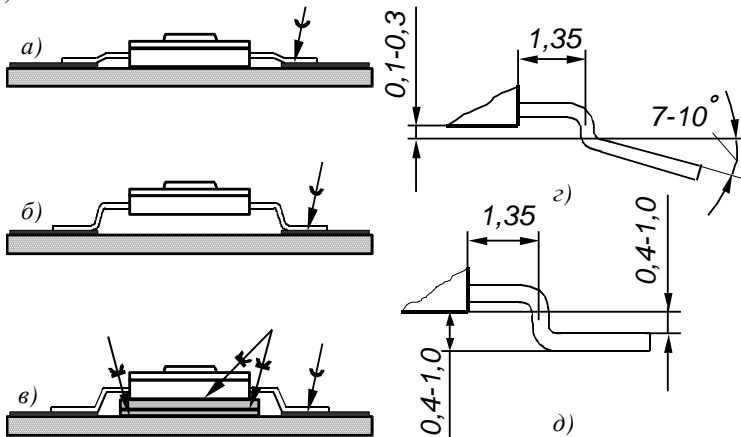


Рис. 14.10. Схема установки на друковану плату корпусів мікросхем з площинними виводами без зазору (а), з зазором (б), з прокладкою (в); з формовкою виводів для установки без зазору (г) та формовкою виводів для установки з зазором

При установці мікросхем з зазором корпус утримується на платі винятково на припаяних виводах. При установці на про-

кладку використовують ізоляційні або металеві матеріали. Як ізоляційні прокладки можуть бути використані смужки тонкого (до 0,3 мм) склотекстоліту або перфорована плівка з вікнами під виводи мікросхем. Прокладку приклеюють до поверхні друкованої плати, а на неї, у свою чергу, приклеюють корпуси мікросхем. Металева прокладка виконує подвійну роль: використовується в якості відводу тепла й екрану. Щоб уникнути виникнення коротких замикань між друкованими провідниками на платі, металеву прокладку ізолюють від поверхні плати спеціальною плівкою.

На рис. 14.11 представлені варіанти установки мікросхем в корпусах з площинними виводами. Крок установки за вузьким боком корпусу 17,5 мм, а за широким – 15 мм.

Площинні виводи утримують корпус мікросхеми тільки за рахунок сил адгезії металізованих контактних площадок з матеріалом плати. Тому що ці сили порівняно невеликі, а маса корпусу може виявитися істотною при великих перевантаженнях, то для пристроїв, що піддаються підвищеному впливу ударів та вібрацій, необхідно передбачати додаткові заходи кріплення корпусів з площинними виводами.

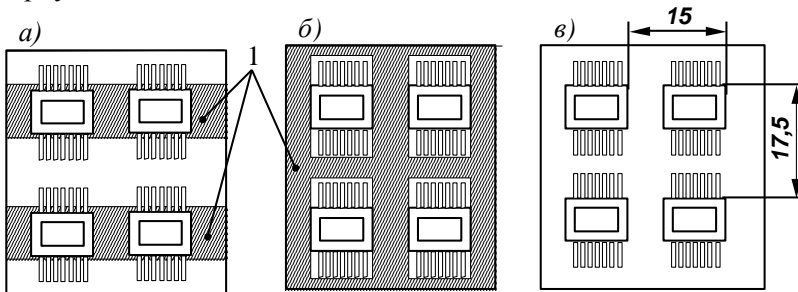


Рис. 14.11. Варіанти установки мікросхем у корпусах з площинними виводами на поздовжніх прокладках (а), на прокладках з вікнами (б) та без прокладки (в)

Варто врахувати також кращу орієнтацію ІС та інших великих компонентів. Так, на двосторонніх платах ІС необхідно розташовувати паралельно з'єднувачам (рис. 14.12, а), а у випадку багатошарових плат – паралельно більшій осі плати (рис. 14.12, б).

Установка та кріплення мікросхем на друкованих платах має забезпечувати їхню нормальну роботу в умовах експлуатації апаратури. Мікросхеми мають бути віддалені від елементів, які при роботі виділяють велику кількість тепла. Їх неприпустимо розташовувати в магнітних полях постійних магнітів, трансформаторів та дроселів.

Зв'язані між собою схеми встановлюють поруч, однак при цьому необхідно стежити, щоб чутливі схеми розташовувалися якнайдалі від схем, у яких велика ймовірність виникнення завад. Швидкодіючі логічні схеми (кола синхронізації, зовнішні логічні кола) примикають до основного з'єднувача, схеми інтерфейсу – до з'єднувача інтерфейсу, а аналогові схеми ізольовані від цифрових (рис. 14.13). Матриці запам'ятовувальних пристроїв мають бути поділені навпіл, а в проміжку між ними варто розмістити схеми об'ємного виведення.

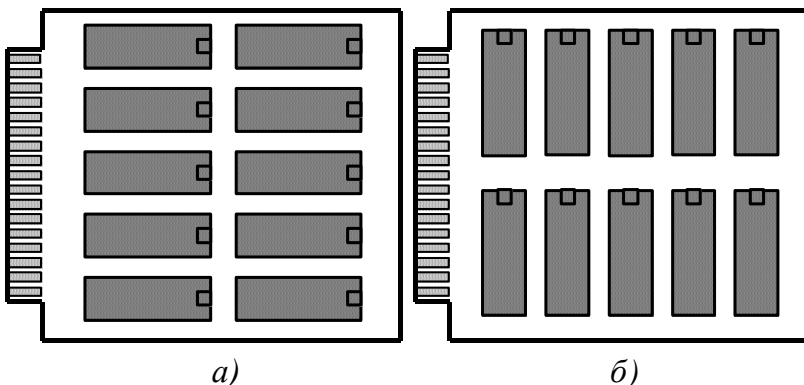


Рис. 14.12. Переважна орієнтація ІС на двосторонніх друкованих платах (а) та багатoshарових друкованих платах (б)

Для правильної орієнтації ІС на друкованій платі мають бути передбачені "ключі", що визначають положення першого виводу кожної мікросхеми (рис. 14.14).

Мікросхеми з відстанню між виводами, кратній 2,5 мм, мають розташовуватись на друкованій платі таким чином, щоб їхні виводи збігалися з вузлами координатної сітки (рис. 14.14). Якщо відстань між виводами мікросхем не кратна 2,5 мм, вони мають розташовуватись так, щоб один чи кілька виводів мікрос-

хеми збігалися з вузлами координатної сітки. На третьому етапі варто продумати схему живлення та кіл заземлення. Є такі можливості:

- використовувати площинні лінії передачі;
- використовувати паралельні шини;
- використовувати площинне заземлення (рис. 14.15).

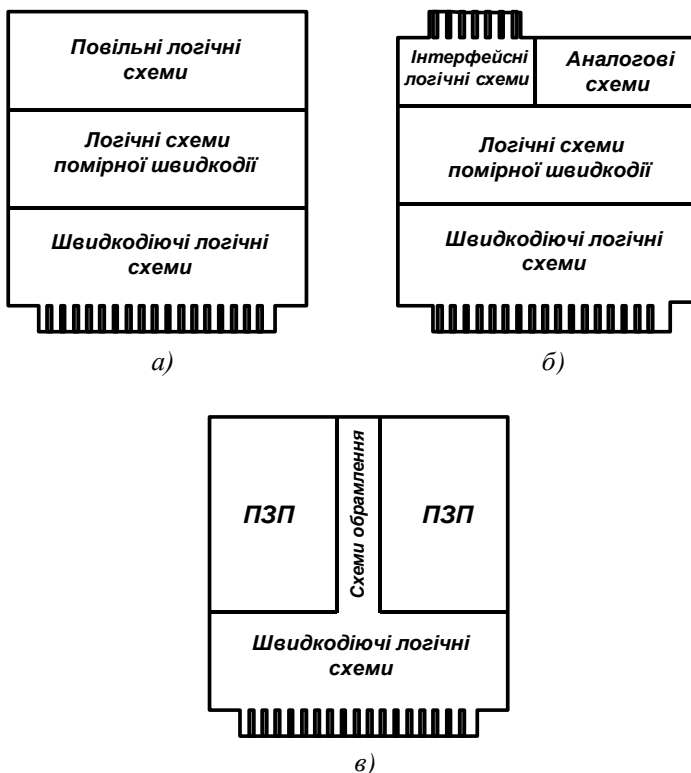


Рис. 14.13. Розміщення компонентів на процесорній платі (а), платі інтерфейсу (б) та платі ПЗП (в)

Копланарні лінії добре зарекомендували себе в двосторонніх друкованих платах. Паралельні шини сприяють зменшенню завад, однак вони повинні мати спеціальну форму. Шини можна розташувати під ІС або паралельно їм. Вони здатні витримувати струм (2,5...15) А, мають розподілену ємність, що не перевищує (0,001...2) мкФ/м та індуктивність (14...35) нГн/м, а

імпеданс між шиною джерела живлення та землею (0,15...5) Ом. Для зменшення завад на шинах заземлення контактні площадки схем заземлення мають бути великими і розташовуватися перпендикулярно шинам.

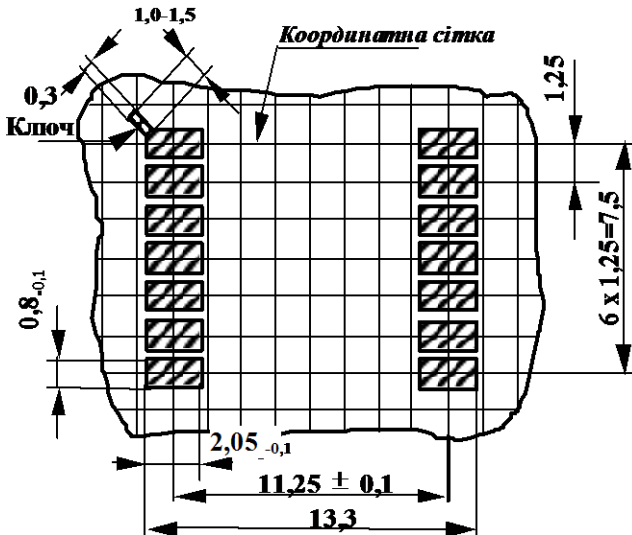


Рис. 14.14. Розмітка посадкових місць для ІС з площинними виводами

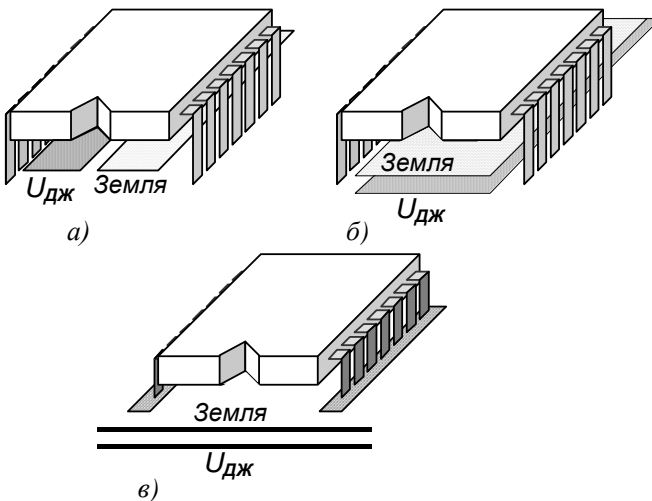


Рис. 14.15. Розміщення шин живлення та заземлення: копланарні лінії (а), паралельне розташування шин (б) та площинне заземлення (в)

У багатошарових друкованих платах можна використувати суцільні шари заземлення і шари живлення; можна також розподілити шари живлення між декількома джерелами. Для більш надійного захисту від завад верхні та нижні шари плати мають бути шарами заземлення або живлення, а сигнальні шари варто розміщати всередині. Шар живлення над шаром заземлення має розподілену ємність $(0,1 \dots 10)$ мкФ/м² і малу індуктивність. В платах типу Multi-wire використовуються тільки шари землі та живлення, а як сигнальні лінії застосовуються тонкі проводи в ізоляції.

З метою розвантаження друкованого монтажу і зменшення кількості шарів плати живлення до мікросхем можна підводити за допомогою навісних шин. Кількість шин може дорівнювати кількості номіналів джерел живлення. Застосовують два способи установки таких шин – перпендикулярно і паралельно площині плати.

На рис. 14.16, а схематично представлений приклад установки навісної шини живлення 1, перпендикулярної площині плати. Живлення 2 до ІС підводиться короткими друкованими провідниками, що відходять від відводів навісної шини, запаяних у металізовані отвори плати. Шина зв'язана відводом з контактом роз'єму, до якого підводиться живлення.

На рис. 14.16, б схематично представлена конструкція, в якій навісна шина живлення 1 паралельна площині плати. Подача живлення 2 до ІС здійснюється індивідуальними відводами. Для кріплення шини необхідно передбачати спеціальні стійки.

Четвертий етап полягає в розміщенні компонентів у кожній схемі. Якщо компонент належить відразу до декількох схем, то його встановлюють поблизу спільної межі. Об'ємні конденсатори і феритові кільця варто помістити якнайближче до виводів живлення і заземлення з'єднувачів. Шунтувальні конденсатори, розв'язувальні кола, а також кола зменшення дугового розряду та викидів напруги мають знаходитися на мінімальному віддаленні (не більше 37 мм) від компонентів, що мають потребу у відповідному захисті. Шунтувальні конденсатори монту-

ються безпосередньо під ІС, що знімає необхідність змінювати топологію плати.

П'ятий етап – вибір розташування контактів силових ліній та ліній заземлення. Потрібно звернути увагу на топологію плат, до недоліків якої відносяться протяжні контури протікання струму, висока паразитна індуктивність між силовими лініями та лініями заземлення. На цьому етапі поширеним є метод зменшення спільних ділянок протікання струмів елементів по шинах живлення. Цей метод полягає в установці додаткових перемичок в шинах живлення і "землі", які зменшують довжину спільних ділянок протікання струмів елементів. На рис. 14.17 наведені три варіанти з'єднання елементів з шиною живлення і "землею".

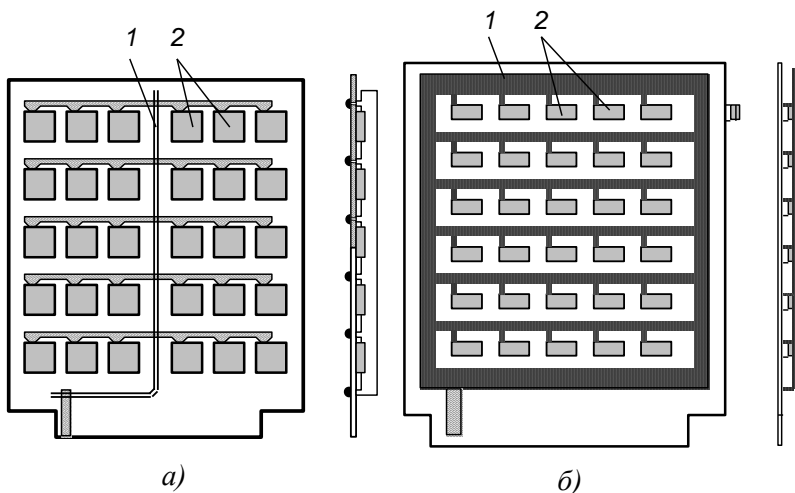


Рис. 14.16. Приклади схеми установки шини живлення (заземлення) перпендикулярно площині плати (а) та паралельно площині плати (б)

У першому варіанті (рис. 14.17, а) перемикання елементу, наприклад E_{12} (зміна споживаного струму схеми), приводить до виникнення паразитної наводки в інших одинадцяти елементах по шині живлення та "землі". В другому варіанті (рис. 14.17, б) ця завада в найгіршому випадку впливає тільки на чотири елементи, а в третьому варіанті (рис. 14.17, в) завада ще більше зменшує рівень за рахунок введення додаткових перемичок. Рі-

вень завад у платах з такою топологією ліній живлення та ліній заземлення близький до того, що спостерігається в дорогих багатопшарових платах.

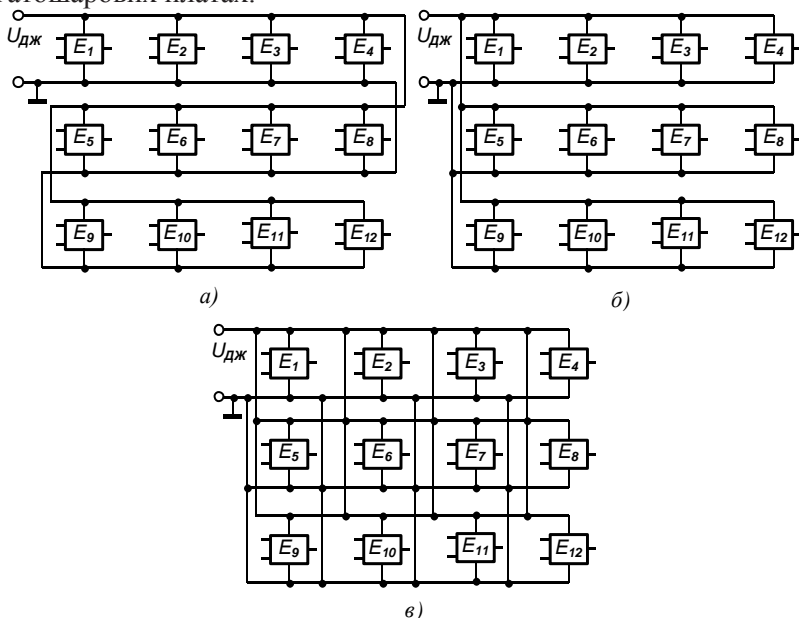


Рис. 14.17. Схеми зменшення спільних ділянок протікання струмів елементів по шинах живлення

Шостий етап – монтаж кіл синхронізації таким чином, щоб вони знаходилися поблизу цифрової землі та подалі від чутливих схем. Добре зарекомендував себе метод, при якому формується пара кіл з прямого і зворотного проводів сигналу синхронізації. Зворотні проводи сигналів синхронізації варто з'єднати з цифровою землею поблизу кожної ІС, яка передає чи приймає імпульси синхронізації. У випадку багатопшарових плат розміщати ці кола необхідно після розміщення шарів живлення і землі. Кола синхронізації мають бути дуже короткими, оскільки імпульси синхронізації та їх гармоніки створюють сильне випромінювання. Один контур площею більше $0,001 \text{ м}^2$ може створювати електромагнітне поле, що набагато перевищує допустимі межі.

Далі звичними методами розташовують інші кола сигна-

льних ліній. Якщо є матриця кристалів ПЗП, адресну шину варто розташовувати в одному напрямку, а шини даних в іншому. Коло до самого молодшого розряду адреси має знаходитися безпосередньо після контакту заземлення. Необхідно прагнути до того, щоб ширина кіл була не більше $1/150$ їхньої довжини, і не допускати зигзагоподібних трактів (рис. 14.18, *а*). Якщо заокруглити вершини кутів, обмежити вигини кутом 45° чи використувати плавні криві (рис. 14.18, *б-г*), то це дозволить підтримувати імпеданс постійним на частотах від 0 до декількох ГГц.

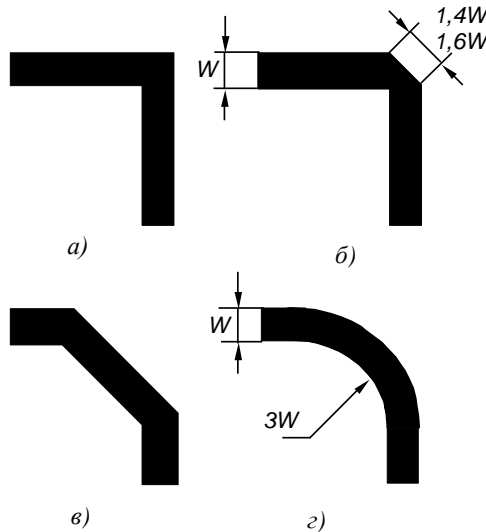


Рис. 14.18. Різновиди вигину контактних площадок на друкованих платах: неприйнятного (*а*) та прийнятного (*б-г*)

На останньому етапі необхідно:

- наскільки можливо розширити кола ліній живлення та ліній заземлення;
- якщо є вільне місце на платі, заповнити його та з'єднати між собою заземлення цифрових схем;
- заповнити вільні ділянки поверхні мідною фольгою та заземлити їх за допомогою кіл або наскрізних отворів.

При цьому утвориться протяжна сітка заземлення, що сприяє зменшенню струмових контурів, імпедансу кіл заземлення і завад, що утворюються за рахунок паразитного зв'язку між

джерелом і землею та між провідниками заземлення.

За необхідності перестроювання рекомендується критично проаналізувати наявний варіант пристрою для виявлення можливостей удосконалення. Вимірюється рівень завад земля-земля між ІС та завад живлення-земля поблизу кожної ІС. Перевіряється за допомогою осцилографа, чи не виникають перехідні процеси у вигляді затухаючих коливань в схемах синхронізації й інших критичних схемах.

Занадто великий час наростання фронтів, перевищення рівня завад живлення-земля та затухаючі коливання на передніх фронтах імпульсів свідчать про недопустимо малі ємності шунтувальних конденсаторів. Амплітуда завад земля-земля більше (150...300) мВ свідчить про велику паразитну індуктивність схеми заземлення. Наявність паразитних затухаючих коливань на спадаючому фронті сигналів означає підвищену паразитну індуктивність кола живлення-земля.

Мікросхеми з'єднують з іншими елементами вузлів і блоків електронної апаратури, як правило, шляхом пайки виводів, тому особлива увага має бути звернена на якість монтажу. В серійному виробництві часто використовують групову пайку і пайку "хвилю". В лабораторних умовах і при заміні мікросхем в експлуатації здійснюють пайку одножилним паяльником.

При розпаюванні площинних виводів ІС одножилним паяльником мають дотримуватися такі вимоги (якщо в ТУ не вказано інше): температура жала паяльника має бути не більше 265 °С, час торкання до кожного виводу не більш 3 с, інтервал між пайками сусідніх виводів 3...10 с (залежно від типу корпусу ІС), відстань від корпусу до місця пайки по довжині виводу має бути не менше 1 мм.

Для мікросхем з штирьовими виводами температура жала паяльника не має бути більше 280 °С.

Контрольні питання

1. Поясніть причини виникнення завад у цифрових пристроях.
2. Яким чином можна підвищити завадостійкість цифро-

вих пристроїв?

3. Які види шунтування використовуються в цифрових схемах?

4. Наведіть схеми узгодження довгих ліній зв'язку для зменшення відбиття сигналів.

5. Назвіть етапи конструювання монтажних плат.

6. Наведіть рекомендації з монтажу ІС з штирьовими выводами.

7. Наведіть рекомендації з монтажу ІС з планарними выводами.

8. Наведіть варіанти розташування шин живлення та заземлення.

9. Поясніть схеми зменшення спільних ділянок протікання струмів елементів по шинах живлення та порівняйте їх ефективність.

ПЕРЕЛІК ЛІТЕРАТУРИ

1. Цифрова схемотехніка електронних систем. Підручник / В.І., Бойко, В.Я. Жуйков, В.М. Співак, А.А. Зорі, В.В. Багрій, Т.О. Терещенко. 3-тє вид. допов. і переробл. – К.: Вища школа, 2010. – 426 с. (гриф надано МОН України – протокол №1.4/18-Г-414 від 14.02.2008 р.). ISBN 966-642-193-3.
2. Макаренко В.В. Моделирование радиоэлектронных устройств с помощью программы NI Multisim / Электронный журнал "Радиоежегодник". – Выпуск: апрель, 2013 (23), с. 141 – 267. – <http://www.rlocman.ru>.
3. Угрюмов Е.П. Цифровая схемотехника: Учеб. пособие для вузов. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2005. – 800 с.: ил.
4. Цифрова та імпульсна схемотехніка. Моделювання та аналіз / В.В. Макаренко, В.М. Співак. Навчальний посібник з грифом Вченої ради НТУУ «КПІ» (протокол . № 6 від 30 червня 2015 р). – К.: НТУУ «КПІ», 2016. – 314 с. іл. ;
5. Рябенкий В.М., Жуйков В.Я., Гулий В.Д. Цифрова схемотехніка: Навч. посібник. – Львів: "Новий світ-2000", 2009. – 736 с.
6. Пухальский Г.И., Новосельцева Т.Я. Цифровые устройства: Учебное пособие для втузов. – СПб.: Политехника, 1996. – 885 с.: ил.
7. Проектирование радиоэлектронных устройств на интегральных микросхемах / Под ред. проф. С.Я. Шаца. – М.: Советское радио, 1976. – 310 с.
8. Программируемые логические ИМС на КМОП-структурах и их применение / П.П. Мальцев, Н.И. Гарбузов, А.П. Шаратов и др. – М.: Энергоатомиздат, 1998. – 158 с.
9. Лебедев О.Н. Применение микросхем памяти в электронных устройствах. – М.: Радио и связь, 1994. – 216 с.
10. Аналоговые и цифровые интегральные микросхемы / Под ред. С.В. Якубовского. – М.: Радио и связь, 1985 – 432 с.
11. Кофанов В.Л. Математичні та схемотехнічні основи цифрових пристроїв: Навч. посібник. – Вінниця: УНІВЕРСУМ-Вінниця, 2005. – 165 с.

Навчальне видання
Підручник

Цифрова та імпульсна схемотехніка.
Моделювання та аналіз

МАЛАНЧУК Євгеній Зіновійович
МАКАРЕНКО Володимир Васильович
СПІВАК Віктор Михайлович
ВЛАСЮК Ганна Григоріївна
РУДИК Андрій Вікторович

Технічний редактор МАЛАНЧУК Євгеній Зіновійович.

2018